Real Time DSP Based Identification System Using Content-Based Imaging Techniques

A Thesis Submitted

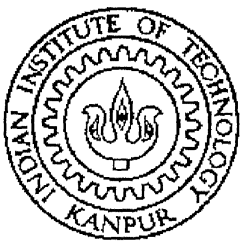
in Partial Fulfilment of the Requirements

for the Degree of

Master of Technology

by

ASHOK KUMAR



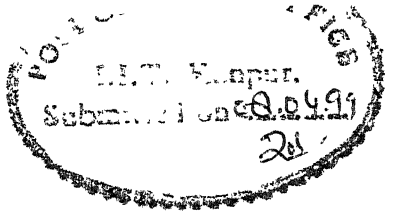
to the

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

April, 1999

25 MAY 1999
CENTRAL LIBRARY
I. I. T., KANPUR
No. A 128048



Certificate

It is certified that the work contained in the thesis entitled "REAL TIME DSP BASED IDENTIFICATION SYSTEM USING CONTENT-BASED IMAGING TECHNIQUES", by Ashok Kumar, has been carried out under my supervision and that this work has not been submitted elsewhere for a degree

(Dr. Sumana Gupta)
Associate Professor
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

April, 1999.

To
My Mother

Acknowledgements

I would like to express my sincere gratitude to Dr. Sumana Gupta for her invaluable guidance and constant encouragement in completing my thesis work. I am grateful to Madam for her suggestions and advises She has been very helpful throughout the duration of the thesis.

I would also like to thank Mr. J. A. R Krishnamooty, Associate Director and Dr O P Nijhawan, Director, I R D.E. Deharadun, for their help and encouragement throughout my M Tech.

I would love to mention the names of my friends Karthik, Sandy, Arindamb, Ritesh, Ganguly, Pratul, Jay, Joyshree, Rohit, Vinay and Berman who have made my stay at IITK, an unforgettable experience.

I specially wish to thank Amit (Neema) for helping me in getting images. It is only appropriate that I mention Shafi who was the final resort for all computer-related problems.

Last but not the least I wish to acknowledge the constant support, blessings and guidance which my family gave me.

Abstract

In this thesis work we propose a new technique for the design and development of an automatic visual identification system The proposed system is implemented using an interconnection of four subsystems: (i) sensing, (ii) data acquisition, (iii) feature(content) extraction and (iv) feature analysis. This system is based on identification of images using content based matching of a query image with those of the database images Query image is the on-line grabbed image by the area scan CCD cameras. Database (off-line) is prepared for all the expected query images by extracting relevant features.

Image histogram (gray level) is used for feature extraction. The computational complexity & storage requirements are reduced by decomposing the histogram using Wavelet Transform. First and second moments of these wavelet coefficients is used as features. The root mean square (*rms*) metric is used to compute the distance between the query image with that of the database images

Although the system is designed to inspect steel slab for surface defects but laboratory evaluation of this system gives excellent performance with general textured and non-textured images also. A setup of high speed Digital Signal Processors is used to keep up with the required real time throughput rates of 1024 Kpixels/sec (1 m wide steel slab moving at the rate of 1 m/sec).

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Quality control is an important aspect of today's highly competitive industrial production. One important way to improve the quality of product is to inspect the product at each level of production cycle. Manual inspection is difficult , time consuming, costly and might impact the effectiveness of human labor because of hazardous environment of industry. High dependency of inspection process on human experience and expertise and also to obtain performance beyond the limit of human ability, has prompted the development of *intelligent programmable vision based system* for inspection. Economic motivation for the use of computer vision is to increase the productivity.

Thus capital formation is often linked to technical innovation, which can produce higher productivity It is our basic postulate that automatic inspection system will raise both labor and general productivity. Limitations and difficulties of automatic web material[1] inspection system development are summarized in next section.

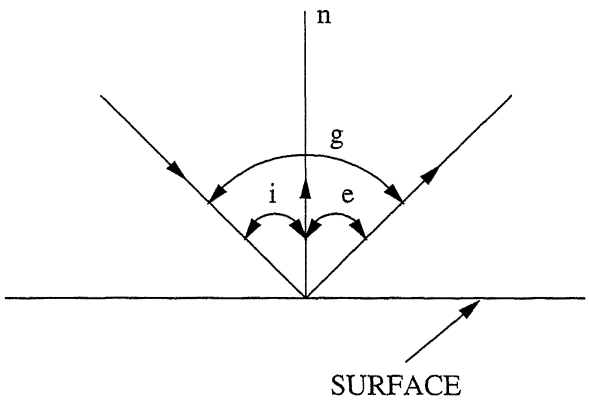
[1]The term *web material* refers to the materials produced in the form of continuous rolls Web processing is used in many segments of industry, e.g. metals, papers, plastics and textiles.

1.1 Issues In Web Material Inspection

Web materials take many forms; however, there is remarkable similarity in the requirements for web inspection technology which cut across the major industrial segments. Typically, materials are homogeneous and discrepancies from homogeneity are interpreted as flaws. Web material inspection primarily relies on two dimensional image understanding in contrast to parts inspection, which is inherently a three dimensional image understanding problem.

Based on generic characteristics, web materials can be categorized in to uniform and texture materials. In both categories, inspection is currently performed either subjectively (by visual inspection) or objectively by destructive or non-destructive testing. Examples of uniform materials include metal, films, paper and various plastics. Identification and classification of defects in these materials is currently the highest technical priority. Texture web material can be divided into regular texture (e.g printed textiles, printed currency) and random textures (e.g. non woven materials). In regular textures any discrepancy from predetermined pattern is viewed as defect. Frequently, these problem require color processing with objective is to evaluate color uniformity and consistency, which is essential in many materials, e.g. wall paper. Random texture materials require grading the overall and/or local texture quality. In some cases the quality of these materials is evaluated through performance tests, e g. by measuring pressure drop across graded surface. In this thesis work we will be discussing the surface defect identification of uniform materials, namely flat cold rolled steel sheets.

Several phenomena is responsible for distribution of light scattered from a metallic surface. The most important effect is the variation of surface normal. The variation of surface normal generally occur on two scales, a fine scale variation representing basic surface roughness and a rough scale waviness resulting from surface irregularities of greater spacing than the fine scale variation. Variation in surface roughness results in variation in surface



normal, hence the reflected power varies. Fig.1. shows the scattering geometry of metal surface. Let E_0 is proportional to power scattered by a smooth plane of area A, then it is given by[6],

$$E_0^2 = \frac{k^2 A^2 I^2}{4n^2 r_0^2} \tag{1.1}$$

Where k = $\frac{2\pi}{\lambda}$,

λ wavelength of illumination,

I cosine of incident angle i,

r_0 distance from the observer to the plane.

Then the reflected power from rough surface is given by

$$P = < \rho\rho^\star > E_0^2 \tag{1.2}$$

Where ρ is the reflection coefficient for scattered power.

Presently, in most steel industry, the hot slab coming out of a caster is cooled to the ambient temperature. An inspector then manually inspects the slab in order to detect surface imperfections. This manual inspection takes lots of time and slow down the overall production rate. Once the slab has been determined to be free of imperfections, it has to be re heated again for further processing. If an automatic inspection system is put, in place of manual inspection process, it will not only increase the production rate but also avoid the intermediate cooling and heating which is necessary for manual inspection, thus we can save lots of energy.

1.1.1 Issues in uniform web material inspection

At present there are commercially available system which can detect the presence or absence of defect at very reasonable cost. However, the problem of defect identification, i e the determination of type of defect is still an open research issue. The major obstacles in solving the identification problem are the following:

- **high data throughput**: A typical web material is 1-3 m wide and moves with a speed of 1-10 m/sec. Consequently, data throughput for 100% inspection (when identifying defect of mm size) is tremendous and cannot be handled by present general purpose hardware.
- **Inter-class similarity and intra-class diversity**: A single class of defect may vary widely in appearance and may have members that closely resemble defect in other classes. Therefore, the structure of the given class in a feature space may be of a very complex nature.
- **Large number of classes** : A typical defect classification problem involves a large number of defect classes; it is not unusual to deal with a few dozen classes.
- **Non availability of adequate imperfection imagery**. Another very significant problem encountered in development of inspection system is the non availability of adequate imperfection imagery for feature extraction and system training. The collection of imagery data is hampered by the hazardous environment in the steel industry This problem is more acute by the fact that the large number of imperfection classes require a large training set of imperfect samples.
- **Dynamic defect populations**: Small changes in production process can result in entirely new classes of defects and a useful classification system should be dynamic with the ability for continuous on-line learning.

The first four items make initial system design very difficult whereas fifth item is a major obstacle in extending the useful lifespan of developed system.

A past study of the feasibility of on-line inspection of a flat rolled steel has been reported by Saridis and Brandin [7]. Recent reviews of automatic inspection as applied to industrial inspection in general, is given in[8]

1.2 Existing Techniques of Defect Identification: A Brief Survey

1.2.1 Edge Detection

In this technique of defect identification, edges of the on-line images are detected. Edges which are present due to noise are discarded by means of filtering operation. After detection of relevant edges, thresholding operation is performed on the image. Presence of edges in the image shows the presence of defect. To identify the type of defect, i.e for defect characterization structural features (shape and size of blobs) of on-line image are extracted. For defect characterization structural features of on-line images are matched with those of data base images. Data base will have information about structural features of all most all known defects class.

The main problem with this technique of defect identification is that it requires extensive learning of different defects geometry over an extended period. Parameters which affects the manufacturing process are very sensitive to industry setup and technical resources also, which results in very high intra-class diversity and inter-class similarity

1.2.2 Profile Matching

In this method of defect analysis profile of on-line images are calculated. It is assumed that profile of non-defective sheets are almost smooth and constant, whereas profile of defective sheet will be zig-zag Nature of profiles will be different for different defects class.

Determination of metal surface profile: For metal surface profile calculation metal surface reflective power is measured in indirect way, and unit of reflective power is *"gray-level"* (0-255) in the digitized images. To make things manageable, it is assumed that the brightest point on in the image has surface normal located on the axis of incident light. Based on this assumption numerical relationship between digitized image and incident angle can be calculated. From the incident angle surface profile can be calculated [6].

This technique is very efficient for defect detection but its performance is very poor in defect characterization.

1.2.3 Smart Sensing and Cortical Projection

In this technique of defect classification *smart sensors* are used for image acquisition rather than the conventional ones. By conventional we refers to sensors with uniform spatial resolution and matrix organization of sensing elements. In this approach, sensing strategy is determined by the defect detection subsystem. Similar to the target tracking system, the inspection system requires the broad view (low resolution image) and based on that determines the presence of potential defects. The presence of potential defect alerts the sensing subsystem through feedback loop. Then, the space invariant sensor foveates on the region specified by the defect detection subsystem to provide the detailed view of potential defect. The switching of sensing strategy is motivated by the desire to reduce data throughput. This sensor

organization is particularly suitable for obtaining a *cortical projection* [13] of an image i.e a geometrical transformation associated with human visual system. Cortical projection simplifies the rotation and scaling effect of object in image plane.

1.2.4 Content Based Technique

Content based imaging technique is increasingly becoming popular in the field of data management and classification because of its ability to reduce the data dimensionality. In content based technique significant features of the image under consideration are extracted. These features are selected in such a way that they form a good discrimination boundary between different class of images and are extracted with ease.

In this thesis we have used content based imaging technique for surface defect identification of flat cold rolled steel sheet. Imagery of sheets are grabbed on-line in synchronism with the speed of sheet. Intensity histogram is used for feature extraction Since the similarity metric used in histogram comparison is of quadratic form, they are computationally very expensive. This computational complexity is the major bottleneck in the design and development of real-time inspection system. To reduce the computational complexity and storage requirements the histogram is represented at varying resolutions using wavelet transform Feature vector is formed by taking first and second moments of wavelets coefficients at all the decomposition levels. Database (off-line) is created by extracting the features and taking their average value.Finally root mean distance is used as a similarity measure. To keep up with the required data throughput a setup of high speed DSP parallel processors is used.

1.3 Thesis Organization

The thesis is organized in the following manner. In chapter 2, we present the feature selection criteria and their extraction details. Inspection system overview and details of the high speed processor setup architecture are presented in chapter 3. Chapter 4 presents the results obtained using the proposed system both for steel sheet images and general textured images. Conclusions and scope for future work is discussed in chapter 5.

Chapter 2

Feature Selection and Extraction

Any object or pattern which can be identified possesses a number of discriminatory properties or features. The first step in any identification or classification process, performed either by a machine or human being, is to select the discriminatory features followed by a method to extract (measure) them It is evident that the number of features needed to successfully perform a given identification task depends on the discriminatory qualities of the chosen features. However, the problem of feature selection is complicated by the fact that the most important features are not easily measurable, or, in many cases their measurement is inhibited by economic considerations.

Feature selection and extraction plays central role in image identification. In fact, the selection of appropriate set of features which take in to account the difficulties present in the extraction process and at the same time result in acceptable performance, is one of the most difficult task the in design of an identification system. Broadly, features can be classified into three categories: (i) physical feature, (ii) structural feature, and (iii) mathematical features. Physical and structural features are commonly used by human beings because these features are easily detected by touch, or by the eye, or any other sensory organs. However, when machines are designed to

identify objects, the effectiveness of these features may be sharply reduced since the capabilities of human sensory organs are difficult to imitate in most practical situations However, machines can be designed to extract mathematical features of object of interest which a human may have some difficulty in determining without mechanical aid Examples of these types of features are statistical means, variances, eigenvalues , eigenvectors and other invariant properties.

In content-based imaging technique the significant features of images are extracted and used to discriminate between the different class of images. Details of feature selection is given in section 2.1. Feature extraction technique is explained in detail in section 2 2. Section 2.3 Explains the similarity technique used in identification.

2.1 Feature Selection for Surface Defect Identification

Our aim is to design a real time inspection system, i.e the system should be able to process and identify the present frame before the arrival of the next frame. Although, by using dedicated high speed parallel processor setup, system throughput can be improved, the inter-processor communication overload and resource limitation, limits the system performance and in practice it happens that the system performance fails to improve by increasing the number of processors used. As the number of processor increases the required resources also increases, economic constraints may fail to meet the demands. Amount of parallelism which can be achieved not only depends on the hardware parallelism but also on the capacity of algorithm to run in parallel. As a specific example, the following points should be kept in mind while selecting features for surface defect identification:

- Steel sheet moving on conveyer belt may experience some amount of

transverse movement, which can result in translation or rotation of the image in the image plane. Feature vectors should be invariant to these effects.

- Feature vectors should be such that, it does not overload the feature extraction process.

Intensity information is an important attribute of image representation. Intensity distribution of an image is generally represented by histogram. Histogram has been found to be a significant feature of an image. As histogram is invariant to translation, rotation and viewing axis, it becomes the ideal choice for feature for on-line identification problem. Other advantage of histogram is, it converts two dimensional image processing problem to a one dimensional distribution problem, i.e, it eases the computational complexity.

2.1.1 Histogram as Feature Vector

The histogram of an image f_n is an N dimensional vector $\{H(f_n, i); i = 1, 2, 3, \ldots, N\}$, where N is number of bins and H(f_n, i) is the number of pixels having intensity i. Given a pair of of images f_n and f_m, the similarity between the two images may be measured using the normalized intersection [10] of their histograms given by,

$$\frac{\sum_{i=1}^{N} min(H(f_n, i), H(f_m, i))}{\sum_{i=1}^{N} H(f_m, i)} \tag{2.1}$$

In this metric intersection, measure is incremented by the number of pixels which are common between the data base image and grabbed on-line (query) image. The measure is finally divided by the total number of pixels in the query image as a normalization factor. It has been shown [10] this metric is fairly insensitive to image resolution, histogram size, occlusion, depth and viewpoint. However, histogram intersection does not take in to account the

perceptual similarity between the different bins of histogram. A metric [12] which take in to account the similarity between the bins is defined in Eq. 2.2.

$$Dist(f_n, f_m) = \sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}[H(f_n, i) - H(f_m, i)][H(f_n, j) - H(f_m, j)]. \quad (2.2)$$

where weights a_{ij} is the cross correlation between bins i and j of histogram. The weights a_{ij} can be normalized so that $0 \leq a_{ij} \leq 1$, with a_{ii}=1, and large a_{ij} denoting similarity between bins i and j, and small a_{ij} the dissimilarity. The two distributions H(f_n) and H(f_m) can also be normalized so that $0 \leq$ H(f_n), H(f_m) ≤ 1 and $\sum_i H(f_n, i) = \sum_i H(f_m, i)$=1.

The distance or more precisely the pseudo-metric shown in Eq. 2.2 is of quadratic form and since histogram is also a high dimensional (N=256) distribution, this distance measure is computationally very expensive. For an image of size X x Y, histogram requires O(XY) addition and O(XY) increments. In addition $O(N^2)$ operations is required to compare a pair of histograms, though this can be improved to order O(N) by some precomputation(e g. by diagonalizing the quadratic form). Moreover because of the presence of a large number of defect class, it is generally not feasible to compute the match measure against every image (O(M) computation if M is the size of the database). Thus, while the histogram has proven to be a good feature, the computational complexity is an inhibiting factor as the size of database increases Hence, the problem at hand is to define a considerably less expensive measure on considerably lower dimensional features so that:

- Overall computational complexity can be reduced in order for the system to work in real time.
- The database can be organized in terms of the lower dimension feature vectors, thereby reducing the storage requirements.

To reduce the computational complexity and storage requirements histogram is decomposed using wavelet transform. In the next section we describe the multiresolution decomposition of the histogram.

2.2 Multiresolution Representation of the Histogram

Multiresolution is widely appreciated in the field of computer vision and image processing, because of its characteristics to decompose and analyze the signal or image, at varying resolutions, which is very similar to human visual system. Histogram of an image is discrete one dimensional distribution of intensity in gray-level space. Multiresolution can be used to analyze this distribution at various resolution. Some heuristic reasons to try out histogram representation at different resolutions are given below :

- Typically, image histogram is sparse with the pixels intensities being concentrated in a few regions of intensity space. This is true even for histogram where intensity axis is coarsely quantized. Even histograms with 256 bins has many bins with very few or no pixels. Consequently, treating the histogram as a simple N-dimensional vector is not an efficient representation especially since the high dimensionality has been the main problem with the computational complexity of histogram comparison techniques. In addition, there is definitely a certain degree of correlation among the adjacent bins of histograms in terms of pixels count. Thus the sparse nature of histogram together with adjacent bin correlation will lead to many coefficients having small or null values in a multiresolution decomposition scheme.

- Multiresolution representation decomposes the signal or distribution in two different components. Coarse component gives information about the pixel concentration in gray-level space whereas contrast information

of an image can be extracted from the detail component.

- Multiresolution representation emphasizes different characteristics of a signal or its distribution at different levels. For example, the coarser resolution can give an idea about the relative amount of intensities in different section of very roughly divided gray-level space while the indices of coefficients with largest magnitudes in lower level can more precisely point to the "*edges*" or to the regions of pixel values concentration in the gray-level space.

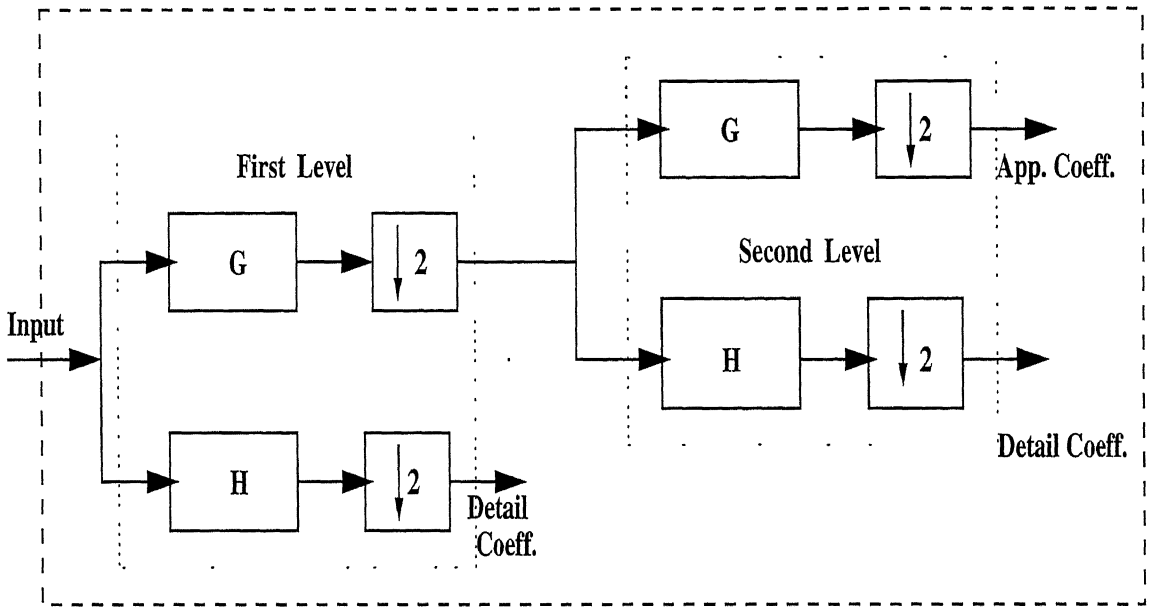
- One of the advantages of using color histogram is its low computational complexity in the feature extraction process. Since multiresolution decomposition is fast and easy to compute, this advantage is not compromised.

2.2.1 The Wavelet Representation

In a multiresolution technique, the signal is approximated at various resolutions. Let A_{2^j} denote an operator which approximates signal at resolution 2^j. Difference of information between signal approximation at resolution 2^j and approximation at 2^{j+1} can be extracted by decomposing the signal or function in a wavelet orthonormal basis. This decomposition defines complete and orthogonal multiresolution representation called "wavelet representation". There exists a function (Wavelet) $\Psi(x)$ such that $(\sqrt{2}^j \Psi(2^j x - k); (j,k) \in Z^2$ is an orthonormal basis of $L^2(R)$ (vector space of measurable square -integrable one dimensional function f(x)). These basis can realize the Haar basis. These basis vectors are localized in both the time and frequency plane. This localization property of wavelet makes it convenient to analyze low frequency content of signal at *long-basis* whereas high frequency content at *short-basis* function. Properties of approximation operator is given in Appendix A.1

Block diagram of a 1-dimensional wavelet transform is given in Fig. 2.1.

Figure 2 1: **Block Diagram of Wavelet Transform**



In this figure, filters G and H are low and high pass filters respectively. The coefficients of these filters depend upon the type of wavelet selected for decomposition. Output of filters are sent to dyadic down samplers. Detail coefficients of first level is stored for extraction of significant features whereas coarse component is applied to next level decomposer for further decomposition This procedure is repeated till the last level of decomposition. Fast discrete one dimensional wavelet transform implementation is explained in Appendix A.2.

2.3 Feature Extraction

Feature vectors of an on-line sheet image are extracted from the histogram of that image. Histogram of on-line images are applied as a input to the multiresolution decomposition filters Filtered outputs of histogram are down sampled by two (2) to get coarse and detail outputs respectively. Details coefficients are stored as it is, and coarse (approximate) component of first

level is applied to decomposer again for next level of decomposition. This procedure is repeated again till the last level of decomposition. Number of levels till which the signal should be decomposed is application dependent and an optimum level of decomposition required can be found experimentally. As a specific example, consider the histogram of N=256 being decomposed up to three levels, the number of coefficients in the first, second and third level of decomposition are 128, 64 and 32 respectively If filters are implemented as a convolution then the number of coefficients for a wavelet having two 2 coefficients are 129, 65 and 33 respectively, i.e data is compressed as the level of decomposition increases.

2.3.1 Feature Vector

As is true in most fields which deal with measuring and interpreting physical events, statistical considerations become important in image classification because of the randomness under which image are grabbed. Especially in industrial application statistical considerations become important because number of parameters which are responsible for intra-class variation and inter-class similarity are very large and the reason of these parameters variation is also not well defined In this defect identification problem, feature vectors are formed by calculating the first and second moments (mathematical features) i e mean and variance of wavelet coefficients. Let V_{hist} denotes the variance of histogram. M_{det1}, V_{det1}, M_{det2}, V_{det2}, M_{det3}, V_{det3} and V_{app3} denotes the mean, variances of first, second, third level detail coefficients and mean of approximation coefficients at level three, then the feature vector is given by Eq. 2.3.

$$FeatureVector = \{V_{hist}, M_{det1}, V_{det1}, M_{det2}, V_{det2}, M_{det3}, V_{det3}, V_{app3}\} \quad (2.3)$$

These feature vectors are used for comparison of the on-line image with those in the data base. Database is also prepared by extracting features of probable defects in the same way but it is an off-line process.

2.3.2 Similarity Metrics

Our basic assumption in feature vector comparison is that the two images which are similar in nature or belongs to the same class will be similar in feature space i.e they will have very small distance in feature space. Let Q_W and T_W be the set of features of two different images, where W is the number of members in each feature vector.

$Q_W = \{V^q_{hist}, M^q_{det1}, V^q_{det1}, M^q_{det2}, V^q_{det2}, M^q_{det3}, V^q_{det3}, V^q_{app}\}$,

$T_W = \{V^t_{hist}, M^t_{det1}, V^t_{det1}, M^t_{det2}, V^t_{det2}, M^t_{det2}, V^t_{det2}, M^t_{det3}, V^t_{det3}\}$.

The root mean square (rms) metric is used to compute the distance between the query image with that of the data base images RMS distance $Dist(Q_W, T_W)$ is given by Eq. 2.4:

$$Dist(Q_W, T_W) = \sqrt{\frac{\sum_{j=1}^{W}(q_j - t_j)^2}{W}} \tag{2.4}$$

Chapter 3

Inspection System Overview

The steel sheets produced in steel industry are susceptible to wide a variety of surface imperfections. A listing of some of the commonly occurring defects is given in table 3 1.

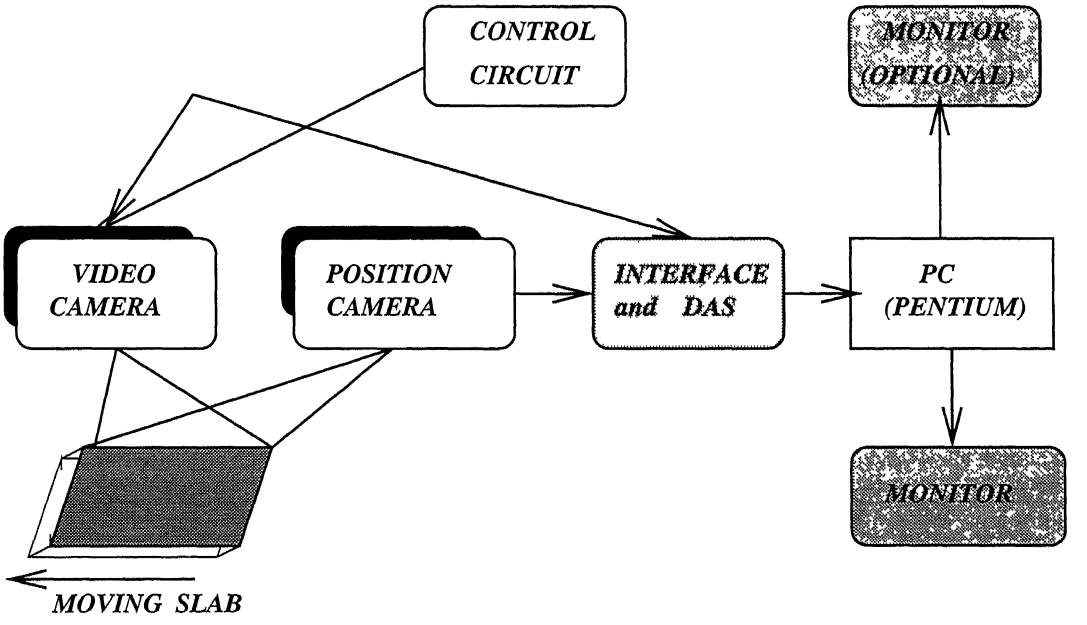
Table 3.1: **Name of Defects**

Coil Break	Roll Mark
Black Patch	Round Scratch
Linear Scratch	Pinch Mark
Color Annealing	Rust in Scale
Iron Particle	Surface Waviness

Our aim is to design a real time visual inspection system for detection and classification of surface defects. The proposed inspection system is shown in Fig. 3.1. The sheet surface is illuminated with an artificial source of light. The camera (video) views the sheet in a direction transverse to the sheet motion. As the sheet moves along the conveyer belt, an analog camera grabs the image arriving in its field of view at a rate which is in synchronism with the movement of sheet in transverse direction. Position camera records the position of sheet from a preset reference position (In some commercially available camera, position can also be recorded together with image acqui-

sition). Video signal (1Vpp) is routed to data acquisition circuit via BNC connector A fast ADC is used to digitize the image. The digitized images are sent to preprocessing circuit of the image grabber card (section 3 2) to remove discrepancy (if any) present due to non-uniform illumination. Finally preprocessed image is stored in video RAM (VRAM) of image grabber card for further processing. Monitor 1. is required to see the on-line identification report whereas Monitor 2. is optional and required only when we want to display the on-line sheet images. Brief description of each of the submodules used in proposed system are given in following subsections.

Figure 3.1: **Block Diagram of Inspection System**



This system will record the date, time and position of each defect, so that these parameter may be taken in to account at a later time e.g. when cutting material. A typical identification report is shown in Fig 3.2.

Figure 3.2: **Identification Report**

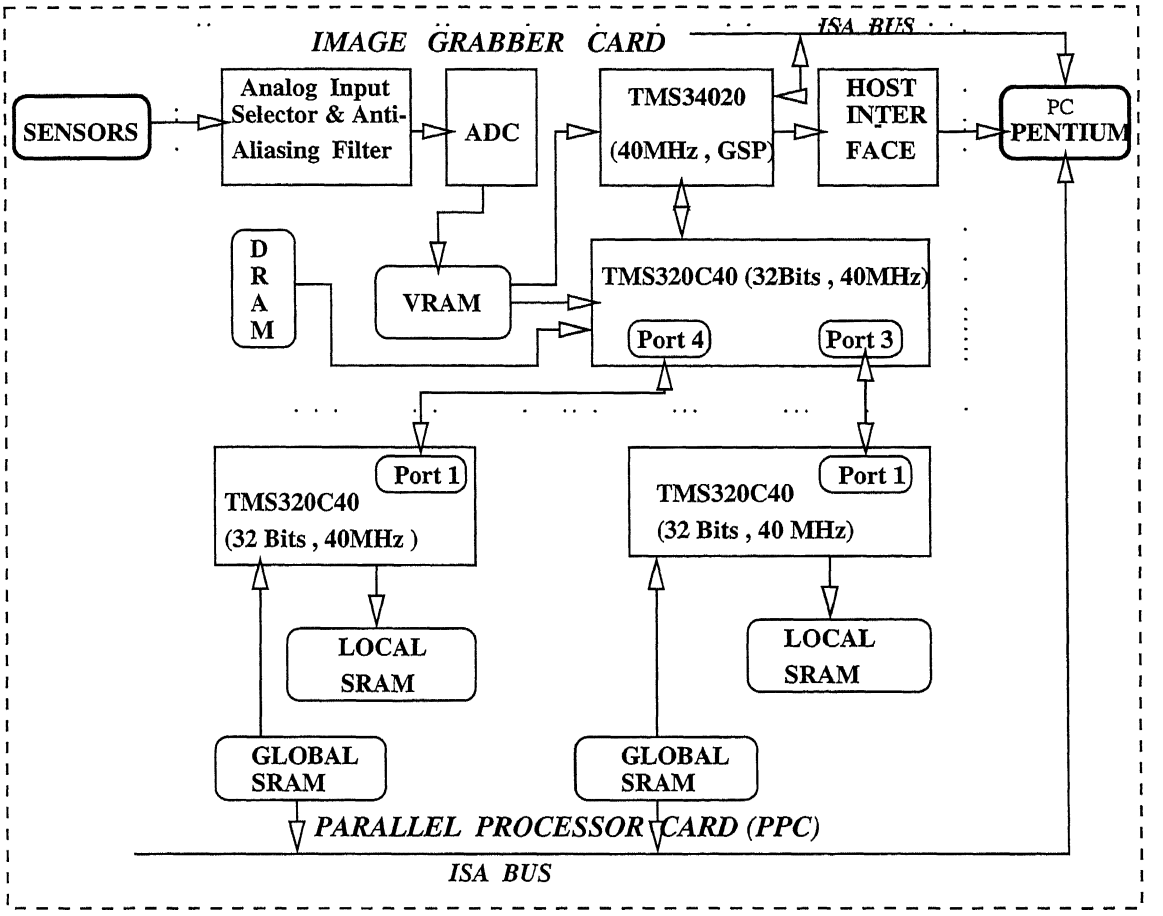
DATE	***12:01:1999***
TIME	***00:01:48***
POSITION	
DEFECT TYPE	***COIL BREAK***

3.1 System Description

In order to illustrate data rates expected in material inspection we consider a specific case For example, consider a 1m wide sheet moving at speed of 1m/sec and the system has to identify defects of 1mm (both in horizontal and vertical direction). Also assume that defect need to be represented by a minimum of 2 pixels in both the horizontal and vertical directions (i.e. spatial resolution of 0.5mm/pixel). It is necessary to place two Charge Coupled Device (CCD) cameras operating at 512 x 512 pixels to cover the cross sheet direction. In order to keep up with the moving sheet, which travels at the rate of 1m/sec, it is necessary to acquire 2 frame/sec. If the CCD camera is operating at 1024 x 1024 pixels, then we need only one camera to cover cross sheet direction and also only 1 (one) frame/sec is required to keep up with the moving sheet, which travels at 1m/sec. Then the defect detection system receives in total 2 x 2 x 512 x 512 = 1024 Kpixels/sec. The need to keep up with this throughput has dictated our design philosophy of both the algorithms and hardware architecture respectively. Block diagram of complete hardware setup is given in Fig. 3.3. Brief introduction of each

module of this system are presented in following subsections.

Figure 3.3: **Block Diagram of System Hardware**



3.1.1 Frame Grabber Card

As shown in Fig. 3.3 frame grabber card mainly consists of four interconnected submodules, they are (i) Analog interface and anti aliasing filter, (ii) Analog to digital converter, (iii) Graphics signal processor (GSP) , (iv) Digital signal processor (DSP), and (v) Video RAM, Dynamic RAM. Video signal of camera are routed to analog to digital converter (ADC) of interface circuit Anti-aliasing filter is used to band limit the signal. Two software selectable anti-aliasing filter are available, one whose corner frequency fixed at

20MHz and second is programmable to have a corner frequency from 1MHz to 10MHz. Depending on the requirements the sampling frequency can be adjusted from 510KHz to 40MHz in increments of less than 10KHz. Digitized image is finally stored in to Video RAM for display and further processing. In case, more than one camera operating simultaneously, image frame are integrated before storing it to VRAM.

Graphics Signal Processor (GSP) TMS34020, is 40MHz advanced processor Its unique array addressing support and efficient manipulation of hardware-supported data types such as pixels and 2-dimensional pixel arrays makes it very useful in imaging and graphics applications. The GSP architecture provides high performance in moving large blocks of data, data management, display control and refresh, image processing, and host communications The TMS34020 is also responsible for all operations within graphics overlay plane. Digital Signal Processor (DSP) TMS320C40, mounted on frame grabber card is a 40MHz, 32 bits floating point processor and can be programmed for specific application.

Frame buffer memory on grabber card is composed of VRAM and DRAM. Video RAM memory can be used for image display and both VRAM and DRAM memory can be used for acquisition and processing. The total memory available on grabber card is divided in to logical partitions called frame buffers. These logical frame buffers are Acquisition Frame Buffer (FFB), Display Frame Buffer (DFB) and Processing Frame Buffer (PFB). The frame buffer dimensions are selected in the driver (Occulus) configuration program. The size of logical frame buffers are same in VRAM and DRAM. VRAM frame buffer memory can be accessed either by GSP or DSP. VRAM frame buffer memory has unique set of frame buffer pointers called (FFB, DFB, PFB) DRAM memory has unique set of frame buffer pointers and can be accessed by DSP only.

3.1.2 Parallel Processor Card

Parallel Processor Card (PPC) is specially designed to meet the requirements of real time processing and embedded system development. It has two very high speed programmable digital signal processors. These processors can be programmed for digital signal processing algorithm as well as other processing algorithms. PPC has local as well as global memory. As shown in Fig. 3.3 the local memory can be accessed by the processor which is connected to it. Global memory is connected to ISA bus and can be accessed by PC as well as the processors. Some of the key features of TMS320C40 are as follows·

- 40-ns and 50-ns instruction cycle time
- High precision and wide dynamic range of floating point (40/32-bits) unit.
- 40/32-bit single-cycle floating point/integer multiplier for high performance in computationally-intensive algorithms.
- Hardware divide and inverse square root support.
- Single-cycle barrel shifter for 0-31 single-cycle right or left shifts for fast bit manipulation.
- Separate internal program, data, and DMA coprocessor busses for support of massive concurrent I/O of program and data throughput, thereby maximizing sustained CPU performance.
- Two identical external data and address busses supporting shared memory systems and high data rate, single-cycle transfers.
- On-chip program cache and dual access/single-cycle RAM for increased memory access performance.
- Six communication ports for high-speed (20 Mbytes/sec asynchronous transfer rate at each port) interprocessor communication.

- Six-channel DMA coprocessor for concurrent I/O and CPU operation, thereby maximizing sustained CPU performance by alleviating the CPU of burdensome I/O.
- Single cycle IEEE floating floating point conversion for efficient interface to IEEE-computable processors.

3.2 Communication Protocol

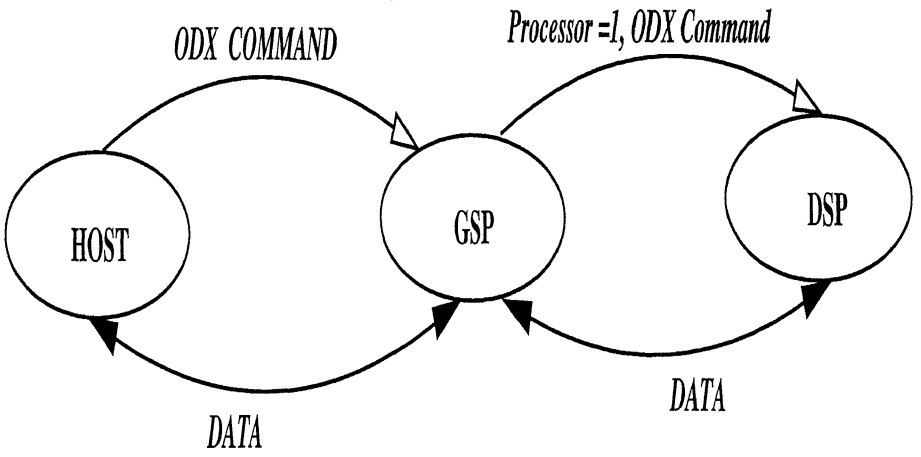
As shown in Fig. 3.3 the proposed system has to work in multi-processor (parallel processor) environment. Processor-to-processor communication is critical in multiprocessor system design. Generally, in parallel processor system, application are divided in to number of dependent and independent task. The task which are dependent requires to share the data of another task operating simultaneously on the other processor This section explains the interprocessor communication protocol of all the processors involved in proposed system design.

3.2.1 Communication Protocol between Host to GSP/DSP of Frame Grabber Card

Usually in parallel processing environment task management is done by application developer. Main program which takes care for system initialization, memory management, data acquisition, synchronizing the start of PPC program (which executes on C40 DSP processor), controlling the frame grabber and setting the processing window size runs on Host. Host program also takes care of the user interface and data transfer to and from the secondary storage device. The data which has to be processed are downloaded in the memory map of processors The DSP/GSP processors mounted on frame grabber card is installed in ISA bus slot of PC (Host). The state diagram shown in Fig. 3.4 explains the communication protocol between Host and

GSP/DSP. This diagram is drawn, assuming only valid ODX command are given, error handling are not detailed here.

Figure 3.4 **Host to GSP/DSP Communication Protocol**



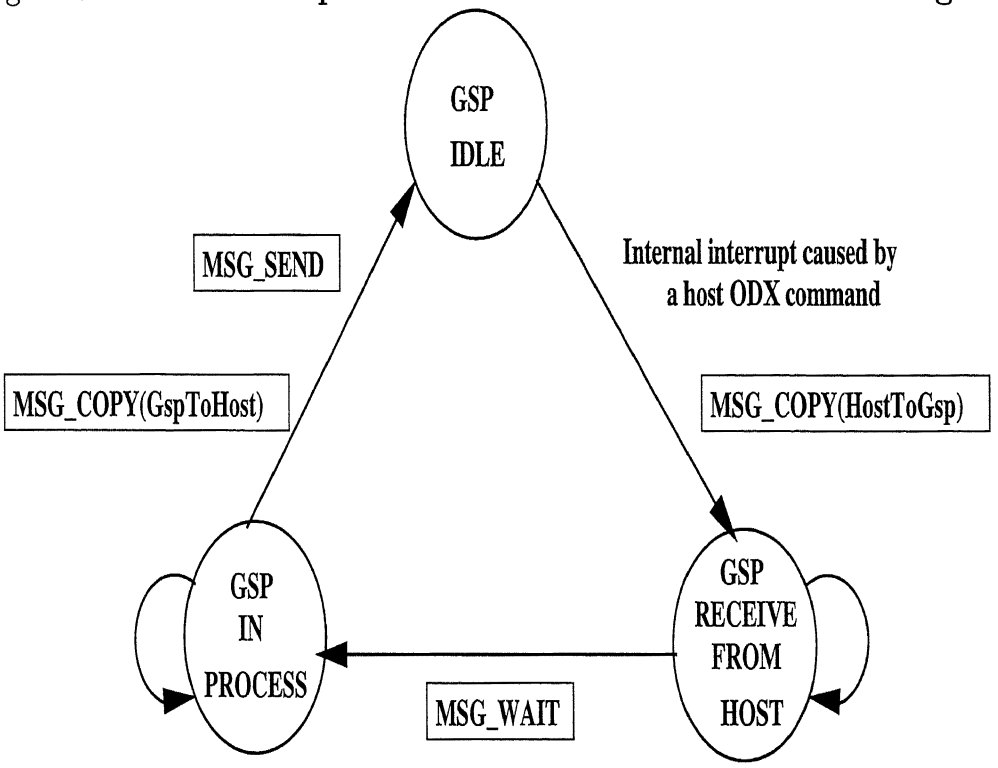
Sequence of Operation for Standard GSP/DSP firmware

- Host Occulus ODX command and data are received by GSP.
- If GSP is active processor it will execute the command.
- If DSP is active processor, the command is passed to the DSP, if the command is within the DSP ODX instruction set. Else the GSP will execute the command without notification or error.

Host-GSP communication state transition diagram

Fig 3.5 shows the communication state transition diagram of Host to GSP. Transition states and the communication functions of the Host-GSP communication are as follows:

Figure 3.5: **Host To Gsp Communication State Transition Diagram**



- A host ODX command is sent to the GSP The GSP goes to receive state, and the ODX command is received. If the command has data, MSG_COPY() is used to transfer the data from host to GSP.

- MSG_WAIT() terminates the transfer from host to GSP (it is not required if MSG_COPY is not used).

- The GSP will process the command. Additional ODX command can be received while the GSP is in the process state if :

  - The ODX command buffer is not full (16 command deep).
  - The ODX command currently in progress or any commands in the buffer do not need to send message or data back to host.

- If there is a data to be returned from a command process, MSG_COPY() is used The command is then terminated by MSG_SEND() which re-

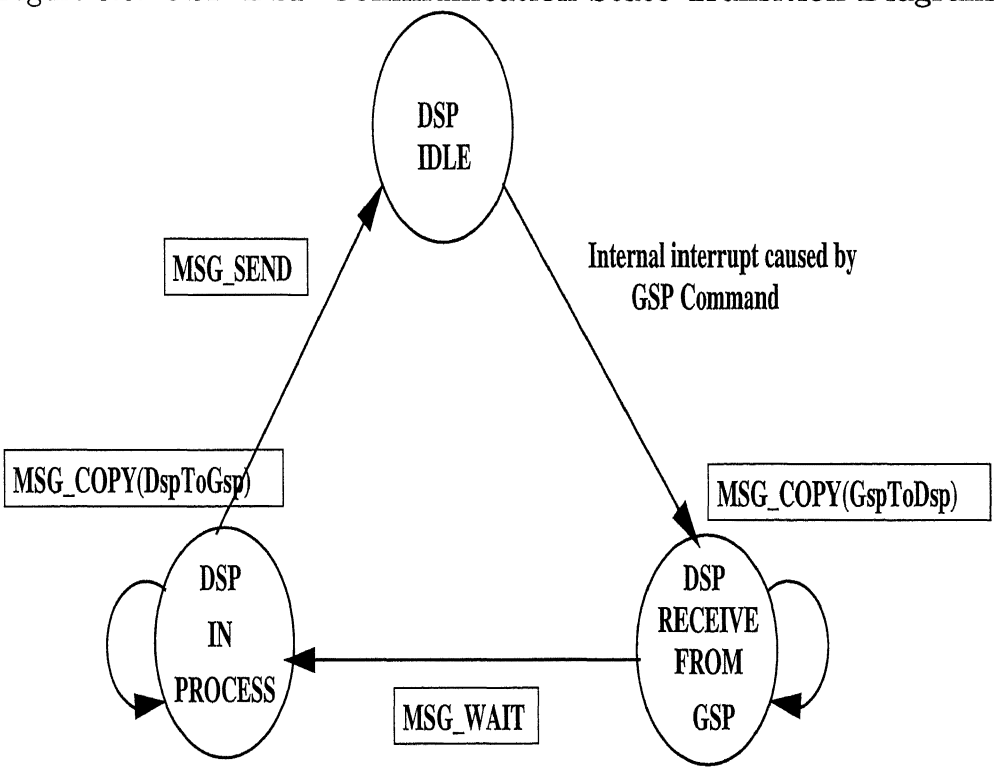
turns a command process value. For example, MSG_SEND(0) for a command executed without error.

- After all commands have been processed and any data or messages sent back to the host, the GSP returns to the IDLE state.

GSP-DSP communication State Transition Diagram

The state transition diagram of GSP-DSP is given in Fig. 3.6. The GSP to DSP communication also follows the conventions of the PC host to GSP communications sequence described above.

Figure 3.6: **GSP-DSP Communication State Transition Diagram**



- An ODX command is sent to the DSP The DSP goes to the receive state. and the ODX command is received. If the command has data, MSG_COPY() is used to transfer the data from GSP to DSP.

- MSG_WAIT() terminates the transfer between GSP to DSP. (It is not required if MSG_COPY is not used)
- The DSP will process the command. Additional ODX commands from GSP can be received while the DSP is in process state if:
  - The command buffer is not full (16 deep buffer).
  - The command currently in progress or any commands in the buffer do not send messages or data back to the GSP.
- If there is data to be returned from a command process, MSG_COPY is used. The command is then terminated by MSG_SEND() which returns a command process value. For example, MSG_SEND(0) for a command executed without error.
- After all command have been processed and any data or messages sent back to the GSP, the DSP returns to the IDLE state.

3.2.2 Communication between Host and PPC

The DSP processor on frame grabber divides the grabbed image and distributes the sub-image to C40s on PPC. The DSP processor on grabber card receives the processed data from the C40s of PPC board. To starts the processing on PPC board Host is required to initialize the the C40s of PPC, set/reset the flags and pass commands as and when required. Communication can be through shared memory where the Host can directly write or read in the global memory location of PPC or using the interrupt scheme where PPC will be interrupting the Host or vice-versa or through status register of PPC processor. An easy way of communication is using shared memory scheme. Communication using shared memory is called as *"Semaphoring"*. The Real Time Library (RTL) provides an easy and convenient way of communicating with, and controlling the PPC. Therefore, by using RTL, the interface with PPC is implemented.

3.2.3 Communication between Frame Grabber DSP and PPC DSP

Communication between DSP of frame grabber and PPC DSP is done through communication port connections by using the callable communication port functions. These callable functions are flexible to use with any DMA channel and any communication port either in transmit mode or in receive mode. Since we have to communicate in both directions i e why we have used DMA coprocessor in *split mode*. *Split mode* transforms one DMA channel in to two DMA channel to make two way communication feasible. The **Primary Channel** is dedicated for reading data from a location in the memory map (external/internal) and writing it to communication port output FIFO. The **Auxiliary Channel** is dedicated to receiving data from a communication port input FIFO and writing it to a location in the memory map A DMA channel in split mode can be used with any communication port; however, read/write synchronization is restricted to signals from its own communication port, in other words, DMA-i can synchronize only with signals coming from communication port i. Appendix B.1 shows typical split mode operation with one communication port.

A split mode communication protocol is given below:

- The primary channel reads word from the address pointed to by the source address register and writes it to temporary register within the DMA coprocessor. It then writes the temporary register value to the output FIFO on the communication port specified in the COM PORT field.
- The auxiliary channel reads a word from the input FIFO on the communication port specified in the COM PORT field and writes it to a temporary register within DMA coprocessor. It then writes the temporary register value in the address pointed to by the destination address register

Chapter 4

Implementation and Laboratory Evaluation

In this chapter we discuss the implementation of the proposed identification system using TMS320C40 parallel processors setup. The system developed is evaluated using different types of textured and non-textured images.

4.1 Implementation Techniques

The proposed identification system mainly consists of two modules: the are *hardware setup* and the *software module.* It is very important in *software design and implementation* to use the available resources to their full capacity. The algorithm discussed in chapter 2 will be used for implementation. The implementation techniques depend on the parallel processor configuration, on the type of processor used and resources available. To finalize the implementation scheme using the available resources (hardware) we measured the identification time required by the system. Lesser the time taken in identification better is the implementation technique.

Although we implemented and tested the overall algorithm module by module by implementing the modules in various ways, in this thesis we con-

sider two slightly different implementation techniques. Fig 4.1 and Fig 4.2 shows the flow chart of the implementation schemes 1 and 2 respectively As shown in Fig. 4 1 the wavelet transform is implemented using on all the three processors by dividing the histogram vectors in three different blocks and using the *master processor* features are extracted. In Fig. 4.2 the wavelet transform is implemented on only two slave processors (TMS320C40) and the coefficients of decomposition are sent parallely to *master processor* for feature extraction Data decimation technique which is part of wavelet transform is not only different for different blocks but also different at different levels of wavelet decomposition. Appendix C.1 and C.2 gives the wavelet transform schemes for the three processors and two processors setup respectively, with *Master Processor* being used for calculating the *Mean* and *Variance.*

4.2 Data Base: An Off-Line Process

Data base preparation is an off-line process. In data base we store the relevant features of images, which are expected during on-line identification. Result of identification depends on the similarity measure values of the online incoming images with those of data base images. Minimum distance value is used to classify the images. Generally, data base feature vector is formed in the same way as the query image feature vector is formed. So, for data base features we have calculated the histogram of the sample images of all the available defective and non-defective classes. These histograms are decomposed using Haar wavelet up to three levels of decomposition. The Mean and Variance of wavelet coefficients at all the three levels of decomposition are used as features. Members of feature vectors are as follows:
{Var. of histogram, Mean of detail coeff. at level 1, Var. of detail coeff. at level 1, Mean of detail coeff. at level 2, Var. of detail coeff. at level 2, Mean of detail coeff. at level3, Var. of detail coeff. at level 3, Var. of coarse coeff. at level 3 }.

Figure 4 1: Flow Chart of Implementation Scheme 1

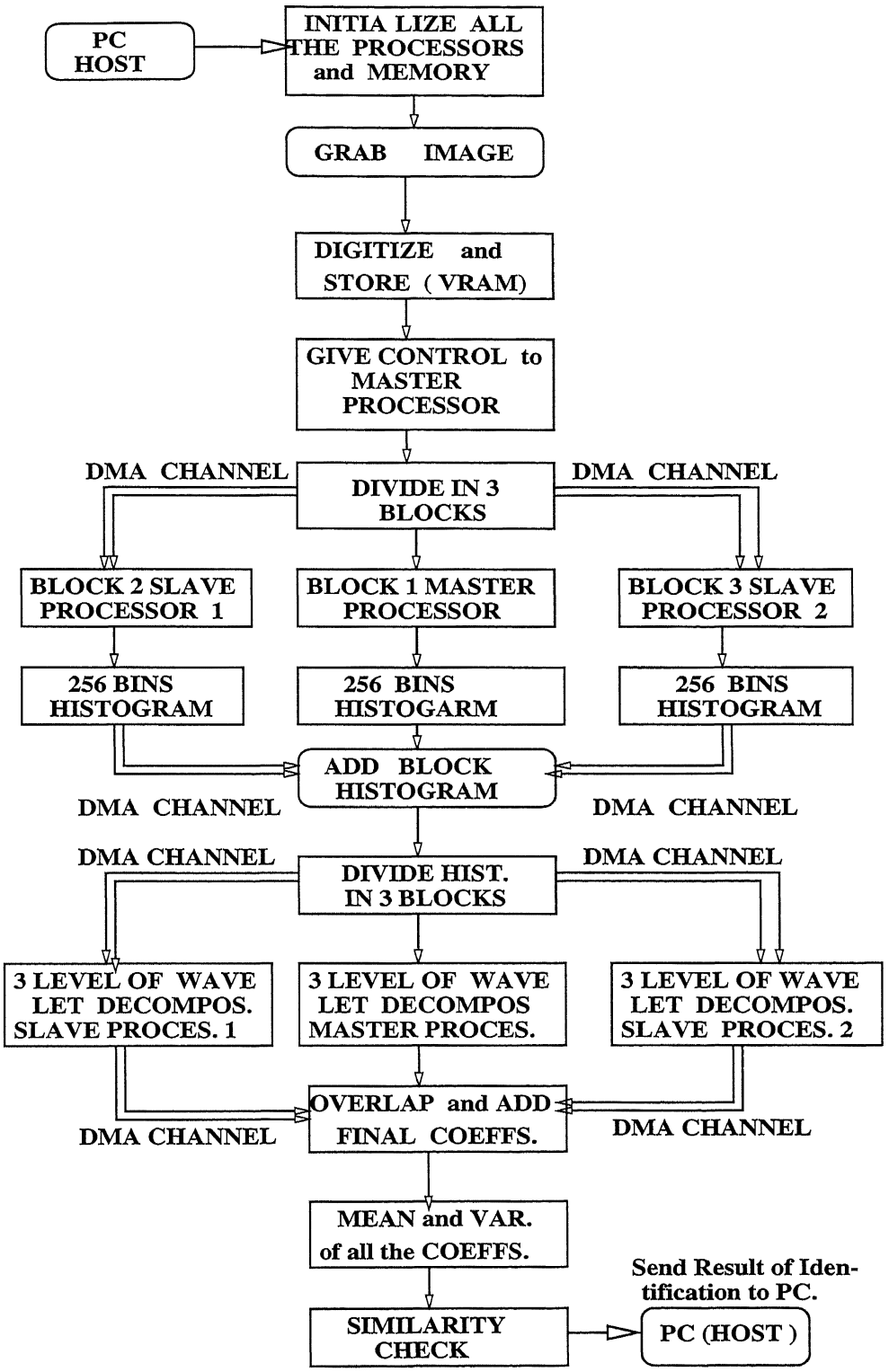
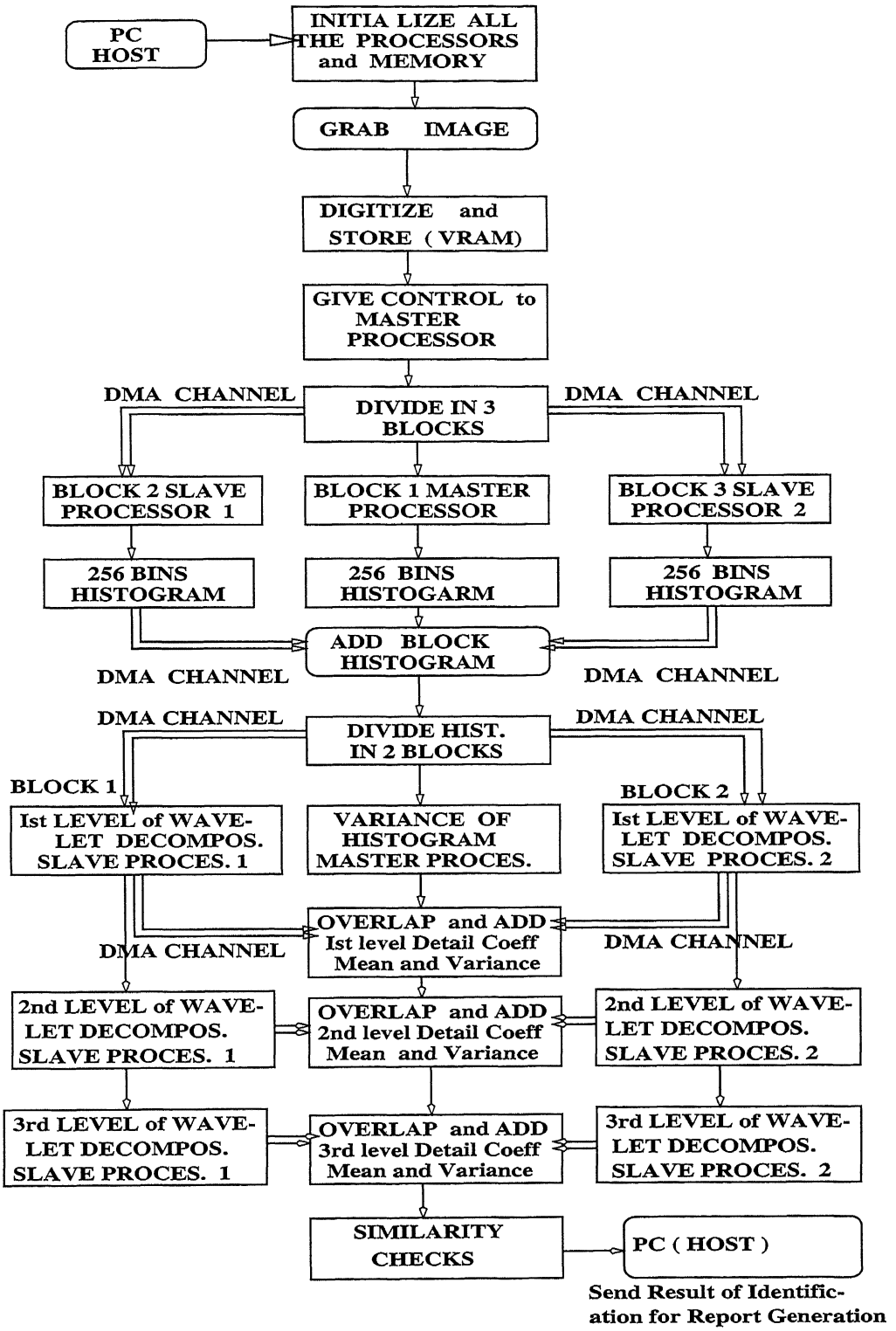


Figure 4 2: Flow Chart of Implementation Scheme 2



Data base feature of particular class is calculated from the average feature vector of sample images feature vectors of that class. The feature vectors for all the classes of image are stored permanently in data base memory locations and compared for similarity with the query images.

4.3 System Evaluation Using Steel Slabs Images

Figs. 4.3 and 4.4 show the images (480 x 360) of steel surfaces having different types of defects. Image histogram is shown on the right of respective images. Intra-class variations of the histogram of some of the commonly occurring class of images are shown in Figs. 4.5, 4.6, 4.7, 4.8 and 4.9. These images were grabbed on-line with the help of area scan camera as the steel sheet moved on the conveyer belt. Feature vector of grabbed images are formed on-line as the steel sheet passes through the field of view of area scan camera mounted in a direction transverse to the sheet motion. To extract the features we followed the same techniques that was used for database, but this time it is an on-line processing, i.e grabbed images are digitized to get the histogram and then it is further decomposed to three levels using wavelet transform. The coefficients of the decompositions are used to calculate the *mean* and *variance*. These features are arranged in the same way as we stored the data base feature vector

To calculate the processing speed, the number of processors required and implementation technique which is best suited for the available hardware, various hardware and software configurations were tested. Table 4.1 shows the relative comparison of number of processors, algorithm implemented and processing speed for a high speed processor setup. From this table 4.1 it is clear that as the number of processors increase the identification time of the system decreases. As we know in a parallel processor environment, inter-processor communication plays a major role, and some times even it becomes

Figure 4.3: **Images of Steel Surface**

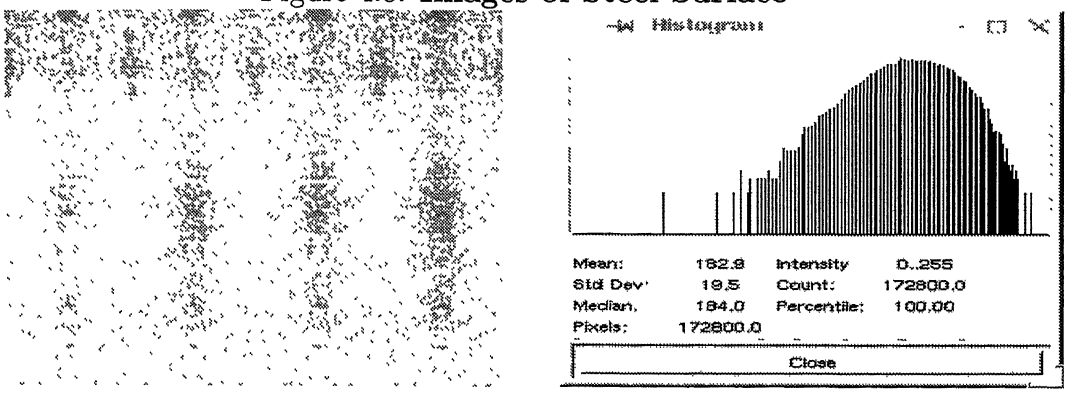


Fig. (a) Non-Defective Steel Slab Image and its Histogram

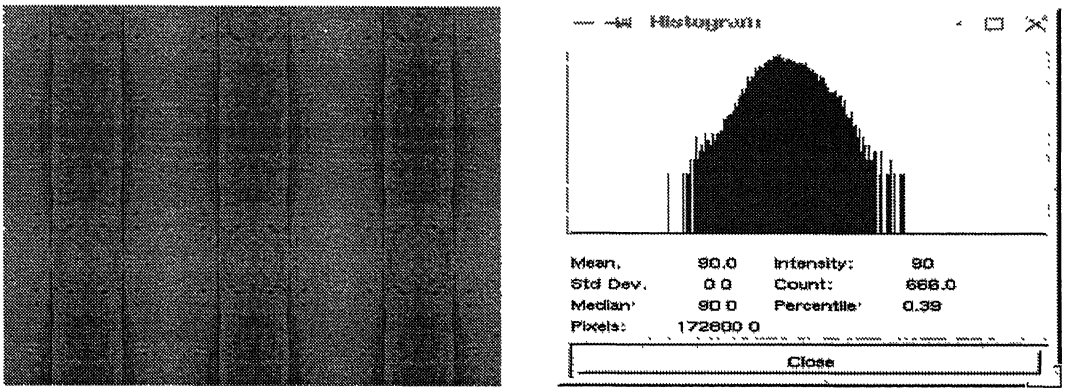


Fig. (b) Coil Break Steel Slab Defective Image and its Histogram

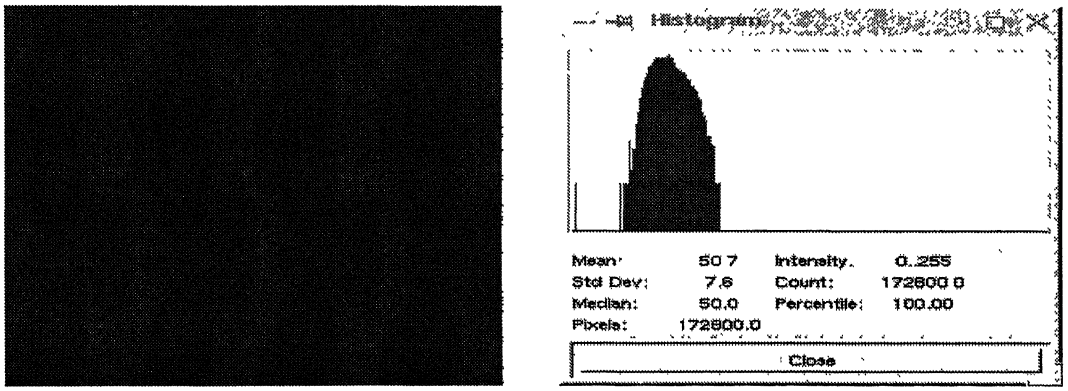


Fig. (c) Black Patch Steel Slab Defective Image and its Histogram

Figure 4.4: **Images of Steel Surface**

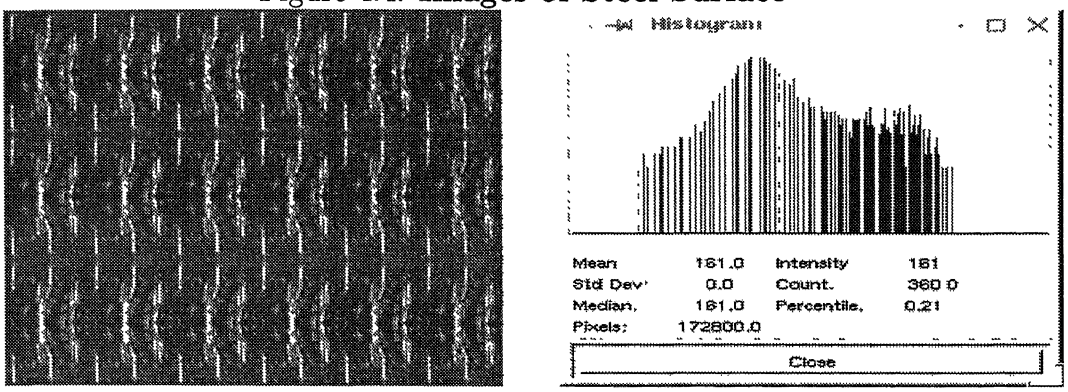


Fig (a) Pinch Mark Steel Slab Defective Image and its Histogram

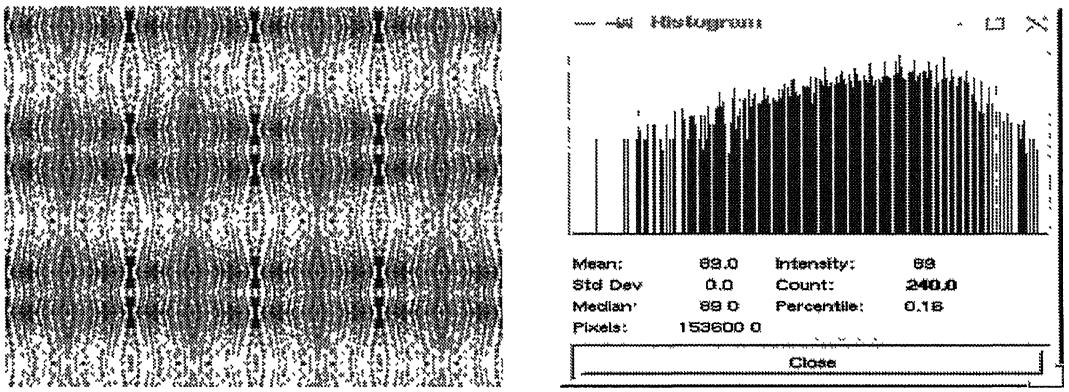


Fig. (b) Roll Mark Steel Slab Defective Image and its Histogram

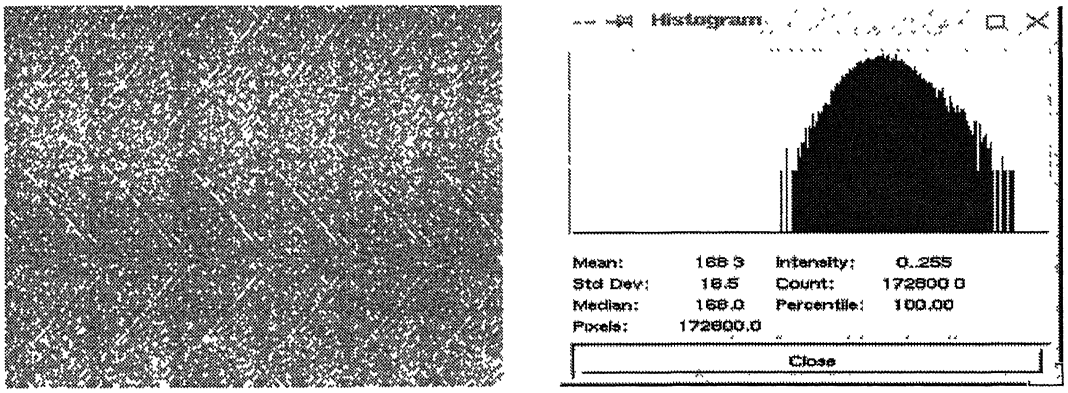


Fig. (c) Scratch Steel Slab Defective Image and its Histogram

Figure 4.5: **Intra-class Histogram Variation of Coil Break Images**

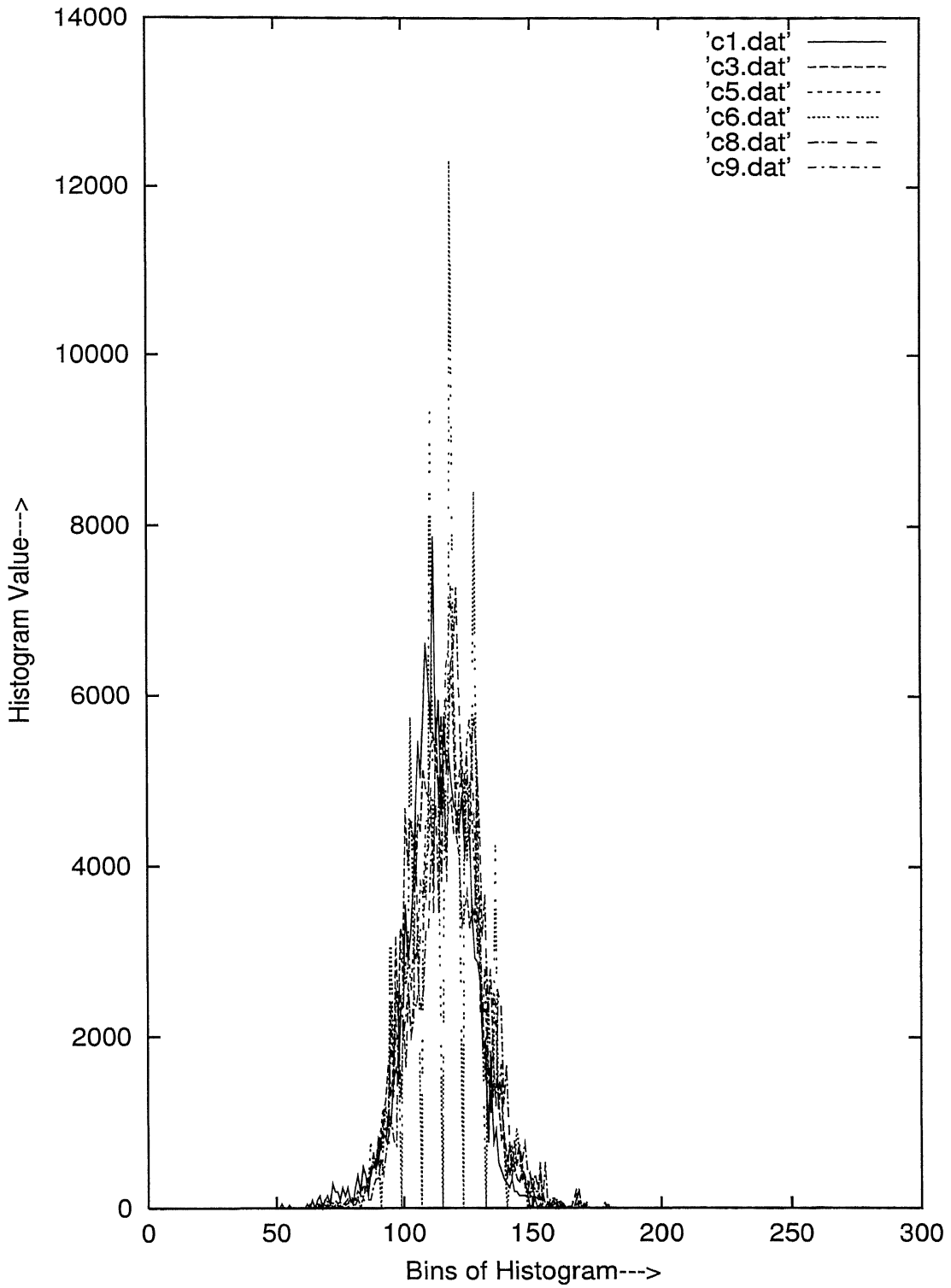


Figure 4 6: **Intra-class Histogram Variation of Roll Mark Images**

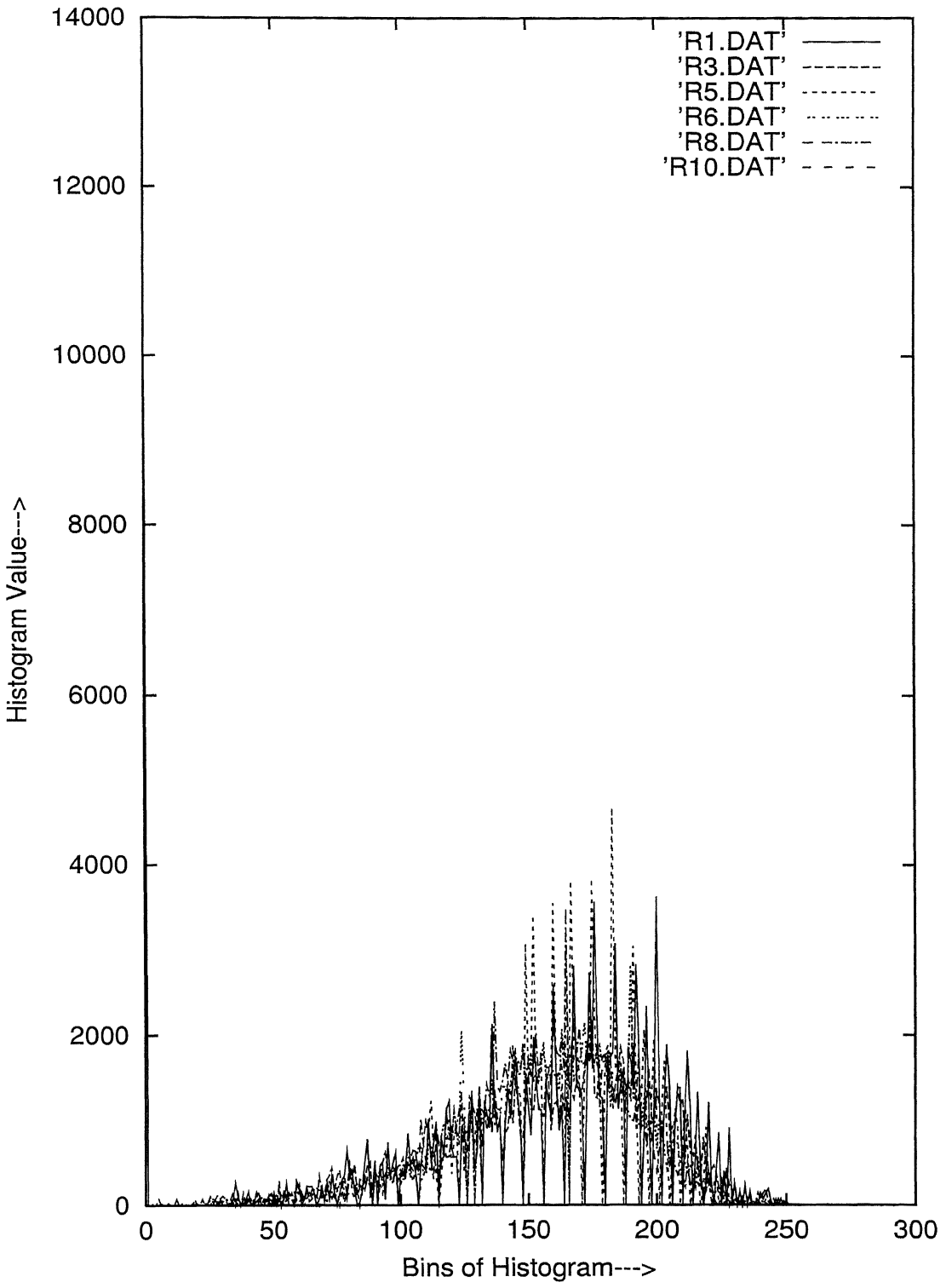


Figure 4.7: **Intra-class Histogram Variation of Pinch Mark Images**

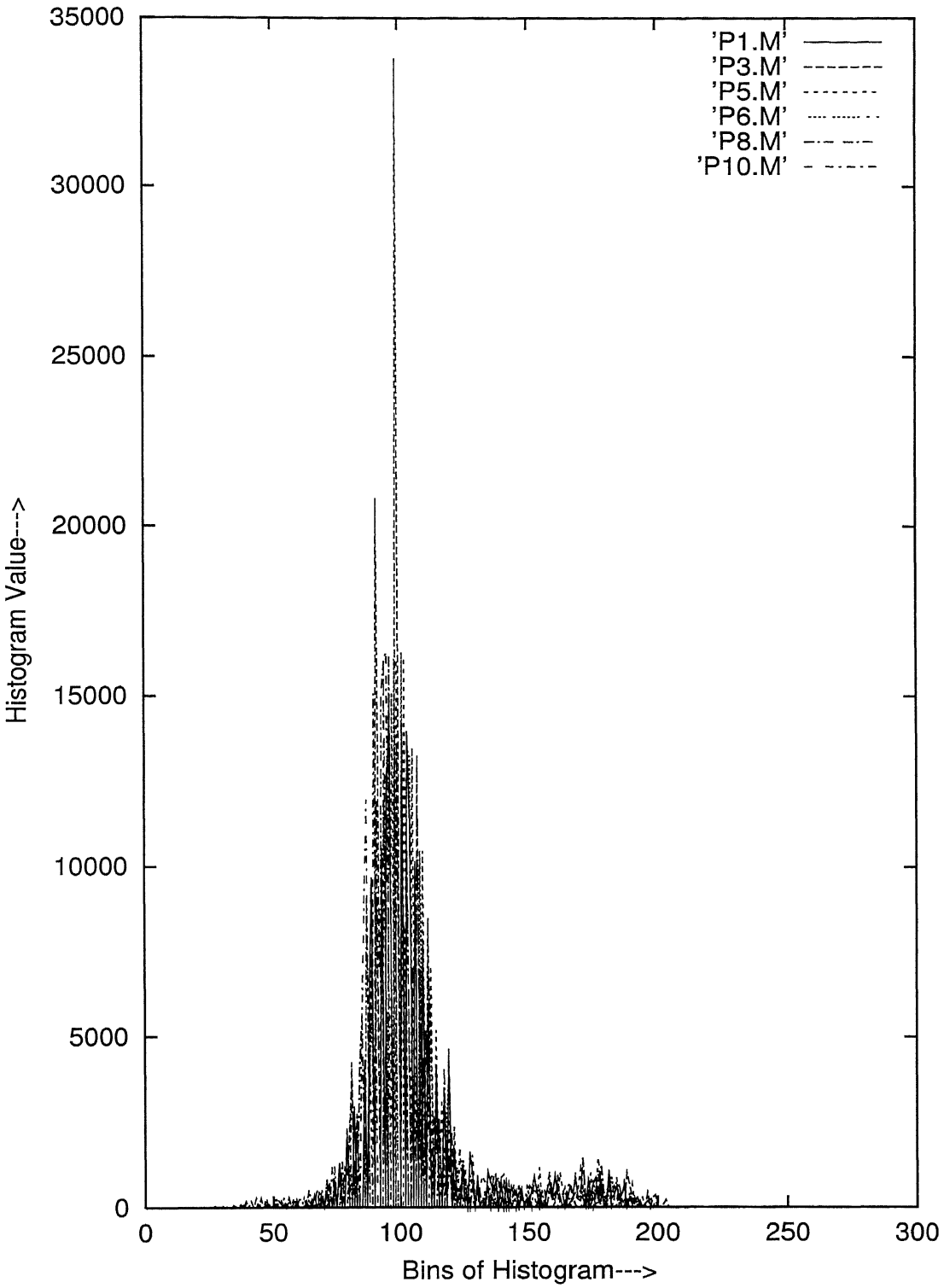


Figure 4.8. **Intra-class Histogram Variation of Linear Scratch Images**

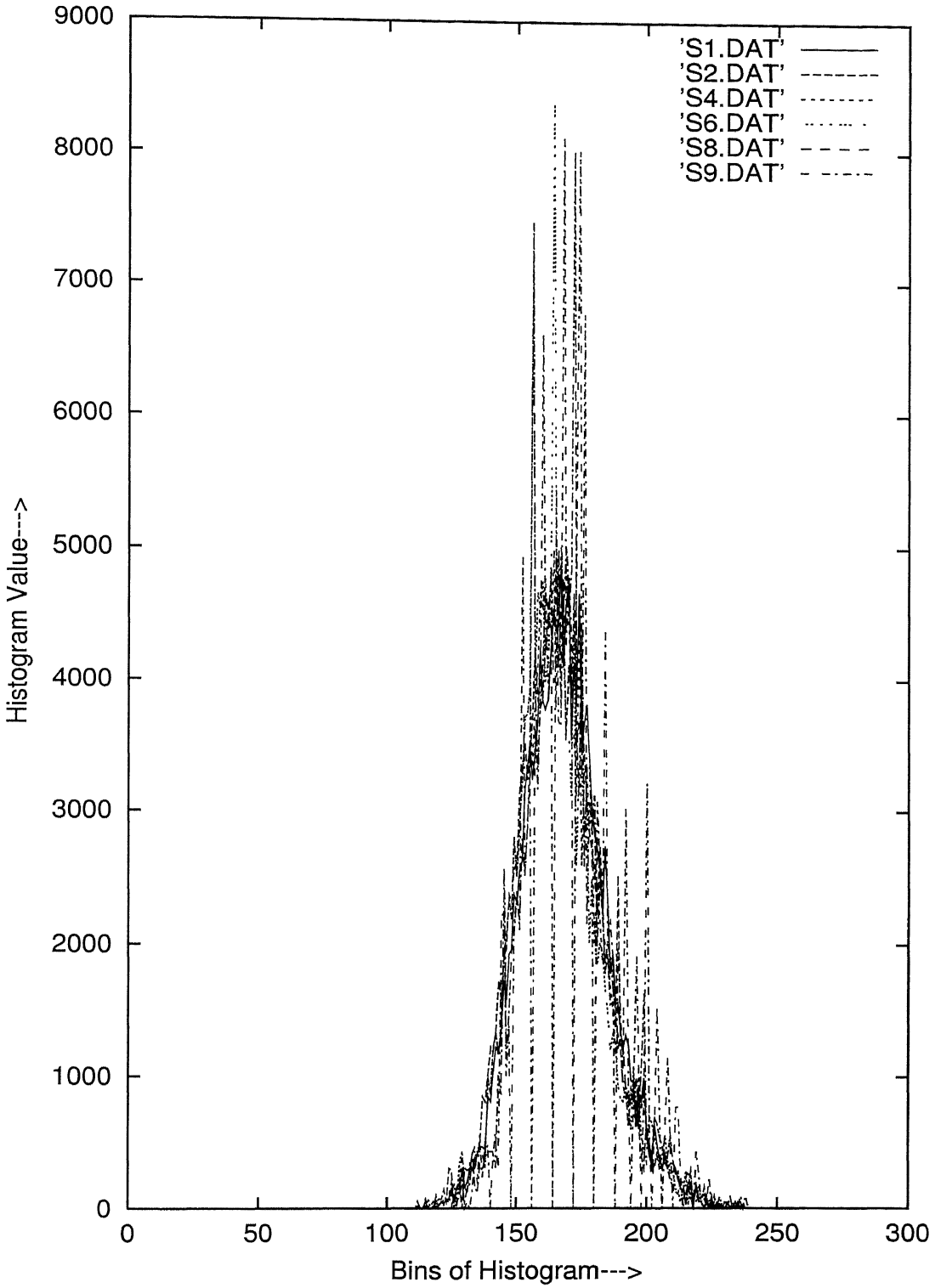


Figure 4.9: **Intra-class Histogram Variation of Black Patch Images**

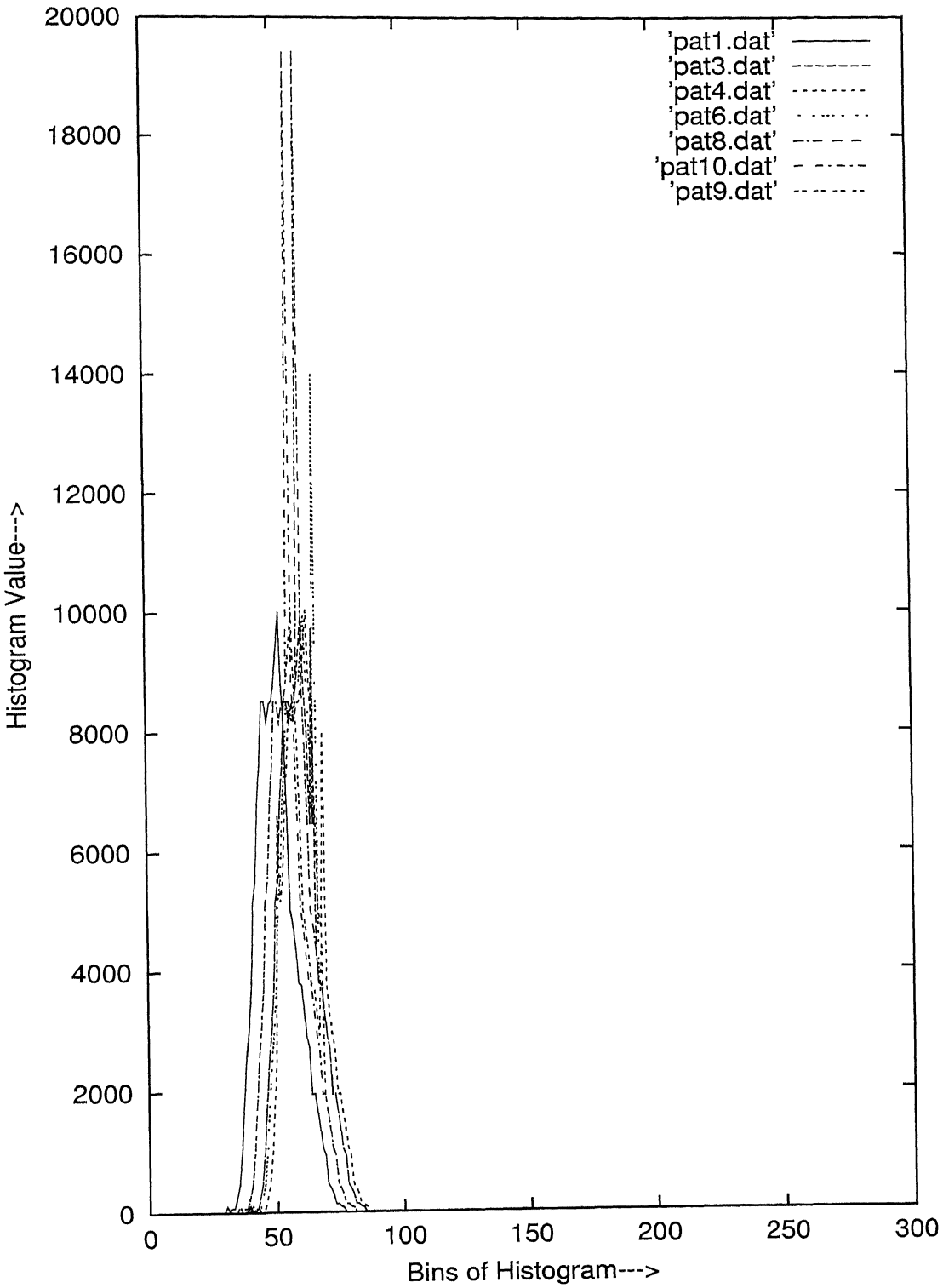


Table 4.1: **Processor and Processing Speed: An Experimental Observation**

No. of processors	Time(ms) required for implementation scheme I	Time(ms) required for implementation scheme II
1	329	329
2	221	218
3	130	117

the major bottleneck in improving the performance of the system. As the number of processors increase the communication overhead also increases. We also note from tables 4.1 and 4 2 that, as the number of processor in-

Table 4.2: **Processors and Percentage Saving in Time**

Implementation scheme	% Saving 2:1	% Saving 3:1	% Saving 3:2
I	32.82	60.48	41.17
II	33.73	64.43	46.33

creases the identification time saving is not in proportion to the increase in number of processors. This saving depends on communication involved and also on the implementation scheme. Implementation scheme II proves to be better with respect to identification time. We used this scheme for further evaluation of system performance.

Although the number of defects occurring in steel industry is large, because of non availability of sufficient number of samples of other defective class, we have evaluated the performance of the system with five defective class of images only. They are as follows:

Class A: Non-defective

Class B: Coil Break

Class C: Black Patch

Class D: Scratch

Class E:	Pinch
Class F:	Roll Mark

In laboratory evaluation of this system we have simulated the on-line industrial inspection process by applying the already stored image frames (defective as well as non-defective and some of them are only used while testing the system). Laboratory evaluation of system gives 100% correct identification. The result of distance metrics for some samples of defective and non-defective classes are given in table 4.3.

4.4 System Evaluation Using Textured and Non-Textured Images

To test the hardware architecture and the identification algorithm we have tested this system with some general textured and non-textured images. These images are classified as follows:

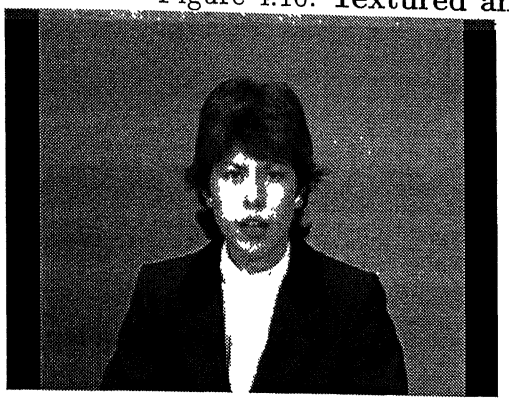
Class A:	Brodatz Texture
Class B:	Garden Images
Class C:	Miss America
Class D:	Salesman
Class E:	Coil Break
Class F:	Black Patch
Class G:	Scratch
Class H:	Pinch
Class I:	Claire
Class J:	Calendar
Class K:	Roll Mark

The images are of size (360 x 288). Fig. 4.10 shows typical samples of images of the new class A new set of data base is prepared by extracting the features

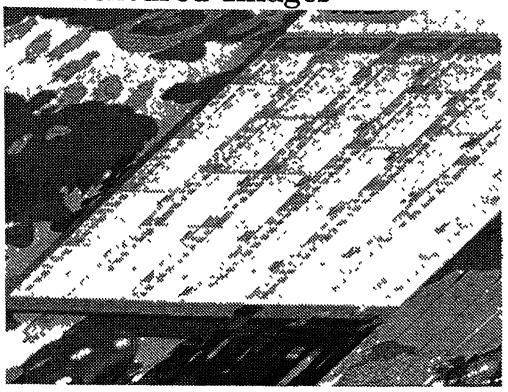
Table 4.3: **Results of System Evaluation Using Steel Surface Defective Images**

Incoming Slab Image	Class A dist.	Class B dist.	Class C dist.	Class D dist.	Class E dist.	Class F dist.
Non-defective	2.3807e+01	2.908e+06	7.1967e+06	1.2575e+06	3.6676e+06	2.5775e+06
Coil break 1	2.2142e+06	8.3471e+05	5.1522e+06	1.0712e+06	3.0629e+06	4.5394e+06
Roll mark 3	2.7809e+06	5.5634e+06	9.8646e+06	3.6743e+06	6.1919e+06	2.0837e+05
Pinch mark 1	3.5014e+06	3.4074e+06	6.0654e+06	3.3991e+06	2.0326e+06	5.3057e+06
Scratch 2	1.2219e+06	2.0171e+06	6.3349e+06	1.5605e+05	3.5961e+06	3.3585e+06
Roll mark 6	2.5400e+06	5.3189e+06	9.6188e+06	3.4305e+06	5.9620e+06	4.3183e+04
Black patch 1	6.9076e+06	4.0154e+06	3.3098e+05	5.9073e+06	4.7532e+06	9.3626e+06
Coil break 9	3.3641e+06	4.0558e+05	3.9294e+06	2.9227e+06	2.8944e+06	5.7576e+06
Scratch 9	1.0084e+06	2.0903e+06	6.3741e+06	3.6782e+05	3.3072e+06	3.2860e+06
Pinch mark	3.6887e+06	2.6961e+06	4.8005e+06	3.3824e+06	5.3887e+05	6.0055e+06
Black patch 6	7.2893e+06	4.4315e+06	2.9732e+05	6.3137e+06	4.9027e+05	9.7481e+06
Roll mark 10	2.7308e+06	5.5345e+06	9.8284e+06	3.6496e+06	6.1056e+06	2.1186e+05

Figure 4.10: **Textured and Non-textured Images**

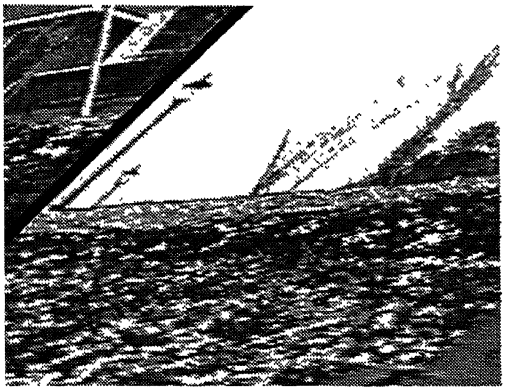


(a) Claire Image

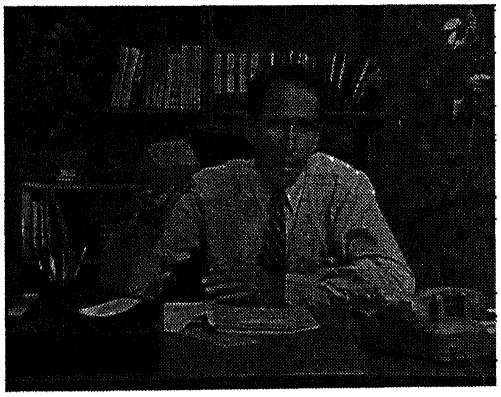


(b) Calender Image

(c) Miss America



(d) Garden Image



(e) Salesman Image

as explained in earlier section and these feature vectors are stored in the data base memory locations. For testing the identification characteristics of the system some of the sample images which are not used for data base preparation are applied as if they are grabbed on-line. The similarity metrics for all these samples are calculated in real time. Result of these measures are shown in table 4.4. The identification time required for these images (360 x 288) using implementation scheme-II with three TMS320C40 DSP processors setup is 73msec.

The result in table 4.4 shows that the proposed system can also be used for identification of general textured and non-textured images.

Since the basic feature extraction depends on the histogram of an image and histogram is highly dependent on illumination level. If illumination level is uniform but not constant then the complete histogram shifts in to intensity space drastically and in the feature space feature vector of that sample image varies quite in amount from their class feature which in turn gives wrong identification result. In laboratory evaluation we increased/decreased the brightness or/and contrast of images using image manipulation program(gimp).

Table 4.4: **Result of System Evaluation Using Textured and Non-textured Images**

Incoming Image	Class A dist.	Class B dist.	Class C dist.	Class D dist.	Class E dist.	Class F dist.	Class G dist.	Class H dist	Class I dist.	Class J dist.	Class K dist.
Salesman	7.9962e+05	6.5367e+05	1.5914e+06	1.2334e+04	1.6155e+06	3.1256e+06	7.9427e+05	1.8770e+06	7.2795e+05	1.2250e+05	2.7890e+05
Claire	1.1818e+06	1.1724e+06	1.2945e+06	7.2993e+05	1.4043e+06	2.8229e+06	7.4246e+05	1.2910e+06	5.7424e+03	7.6545e+05	8.7138e+05
Miss America	2.3319e+06	2.2249e+06	2.4951e+04	1.6019e+06	3.6892e+05	1.6046e+06	8.3527e+05	8.6853e+05	1.2857e+06	1.6961e+06	1.8576e+06
Coil Break	2.4824e+06	2.3624e+06	4.1175e+05	1.7213e+06	1.0415e+05	1.4213e+06	9.1719e+05	1.0516e+06	1.5020e+06	1.8267e+06	1.9867e+06
Roll Mark	5.9194e+05	4.3945e+05	1.7986e+06	2.0261e+05	1.8292e+06	3.3389e+06	1.0073e+06	2.0492e+06	8.3826e+05	1.0111e+05	6.4170e+04
Brodatz	1.2392e+05	1.7411e+05	2.3542e+06	7.7925e+05	2.3977e+06	3.9044e+06	1.5759e+06	2.5162e+06	1.2488e+06	6.7408e+05	5.1432e+05
Black Patch	3.8076e+06	3.6958e+06	1.5189e+06	3.0559e+06	1.4287e+06	8.8406e+04	2.2500e+06	1.7579e+06	2.7457e+06	3.1598e+06	3.3204e+06
Calendar	7.0932e+05	5.6391e+05	1.6714e+06	8.4170e+04	1.7044e+06	3.2136e+06	8.8305e+05	1.9391e+06	7.5342e+05	2.7875e+04	1.9011e+05
Scratch	1.5702e+06	1.4437e+06	8.4517e+05	8.0260e+05	8.2592e+05	2.3366e+06	4.2130e+04	1.2934e+06	7.7370e+05	9.0831e+05	1.0680e+06
Pinch Mark	2.6740e+06	2.6601e+06	1.0589e+06	2.1289e+06	1.2368e+06	1.8239e+06	1.5237e+06	4.3169e+05	1.4905e+06	2.1982e+06	2.3328e+06
Garden	2.4113e+05	2.3139e+04	2.2491e+06	6.6373e+05	2.2903e+06	3.7994e+06	1.4686e+06	2.4575e+06	1.1945e+06	5.5818e+05	3.9847e+05

Chapter 5

Conclusions and Future Extensions

In this thesis work we have proposed an implementation of a real time identification system using high speed parallel DSP processors. The software module is mainly written (application program) in 'C' language, except for the communication and initialization modules which are written in *assembly* language.

5.1 Conclusions

- Features are extracted from intensity histogram at various resolutions. This not only reduces the computational complexity but has proved to be a robust method for environments where image translations and rotations can not be avoided.

- Since the system works in real time, identification time required should be well within the specification It is experimentally measured that the proposed system will be able to identify images of size 1024 x 1024 in less than 1 sec. Results of laboratory evaluation of this system with textured as well as non-textured images are found to be acceptable.

- As, histogram is invariant to translation, rotation etc., the result of identification does not depend on these parameters. However, histogram is very sensitive to illumination, and maintaining an uniform and constant illumination is a crucial requirement.

- Result of the laboratory evaluation of this system shows that the system can be used for identification of faces, finger prints etc., for which a data base is prepared and feature vectors stored. We can also think of using this system as *smart sensors* for *smart robot.* There are wide application of this system, and one can think of using it as an *intelligent system* in a well disciplined environment

- Performance of the system would be better in terms of processing speed if *PCI* bus architecture was used in place of *ISA* bus architecture for purpose of communication between frame grabber card and PC (Host).

5.2 Scope for Future Works

- Once the system performance and algorithm is found to be satisfactory in industrial environments, the processing speed can be improved by converting and optimizing the 'C' routines in assembly language and finally putting the overall code in EEPROM so that when system is switched on, it will start its specified work without user request.

- This system does not specify the exact positions of defects in a particular defective image frame. It only identifies the whole frame as defective or non-defective. With the help of block histogram (spatial segmentation of image frame and their histogram) at various resolutions we can specify the positions of defects exactly in that image frame

- Present system is not being tested for more than one defect occurring in a particular image frame. With the help of spatial segmentation and

block histograms at various resolutions we can find the feature vectors for all the blocks separately and store it in the data base for similarity measurements. By segmenting the on-line grabbed images in the same way as the data base is prepared, we can identify more than one defect in the grabbed on-line images.

- Performance of the system will be better if we can extract the features from color histogram at various resolutions in place of extracting the features from intensity histograms. This requires a *color camera with color image grabber card.*

- Processing time can be improved by decomposing the identification technique into two levels. Defect detection algorithm can be implemented at the first level. If defects are present, only those images are sent to the classification module.

- In place of using software routines for defect detection we can use smart sensors for defect detection followed by software routines for classification. This improves the system performance drastically because the hardware is always faster, but reduces the portability of system.

CENTRAL LIBRARY
I. I. T., KANPUR
Acc. No. A 128048

Appendix

Appendix A.1

1 Approximation operator A_{2^j} is linear operator. If A_{2^j} is approximation of some signal f(x) ($f(x) \in L^2(R)$) at resolution 2^j, then, A_{2^j} is not modified if we approximate it once again at the same resolution 2^j. i e $A_{2^j} \circ A_{2^j} = A_{2^j}$. Thus A_2^j is a projection operator on a particular vector space $V_{2^j} \subset L^2(R)$ The vector space V_{2^j} can be interpreted as set of all possible approximations at the resolution 2^j of functions in $L^2(R)$

2. Among all the approximated function at resolution 2^j, A_{2^j} is most similar to f(x). i.e, $\forall g(x) \in V_{2^j}$, $(modg(x) - f(x)) \geq \|A_{2^j}$, Hence, the operator A_{2^j} is an orthogonal projection on the vector space V_{2^j}.

3. The approximation of a signal at resolution 2^{j+1} contains all the necessary information to compute the same signal at resolution 2^j. This is a causality property. $\forall j \in Z, V_{2^j} \subset V_{2^{j+1}}$

4. An approximation operator is similar at all resolutions. The spaces of approximated function should thus be derived from one another by scaling each approximated function by ratio of their resolution values,

   $\forall j \in Z, f(x) \in V_{2^j} \Leftrightarrow f(2x) \in V_{2^{j+1}}$

5. The approximation A_{2^j} of a signal f(x) can be characterize by 2^j samples per unit length. When f(x) is translated by a length proportional to

2^{-j}, A_{2^j}f(x) is translated by the same amount and is characterized by the same samples which have been translated.

6 When computing an approximation of f(x) at resolution 2^j, some information about f(x) is lost. However, as the resolution increases to $+\infty$ the approximated signal should converge to original signal. Conversely as the resolution decreases to zero, the approximated signal contains less and less information and converges to zero.

Appendix A.2

This appendix describes the algorithm for computing fast discrete wavelet transform. Let G and H are decomposition low and high pass filters respectively. Let $S^d_{2^j} f$ is digitized image at resolution j. We denote (A * B) as convolution of discrete signal A and B. At each scale 2_j, it decomposes discrete signal $S^d_{2_j} f$ in to coarse signal $S^d_{2^{j+1}} f$ and detail signal $W^d_{2^{j+1}} f$.

j=0
while (j $\leq$ J) ; J :maximum number of level of decomposition

$W^d_{2^{j+1}} f = S^d_{2^j} f * H$

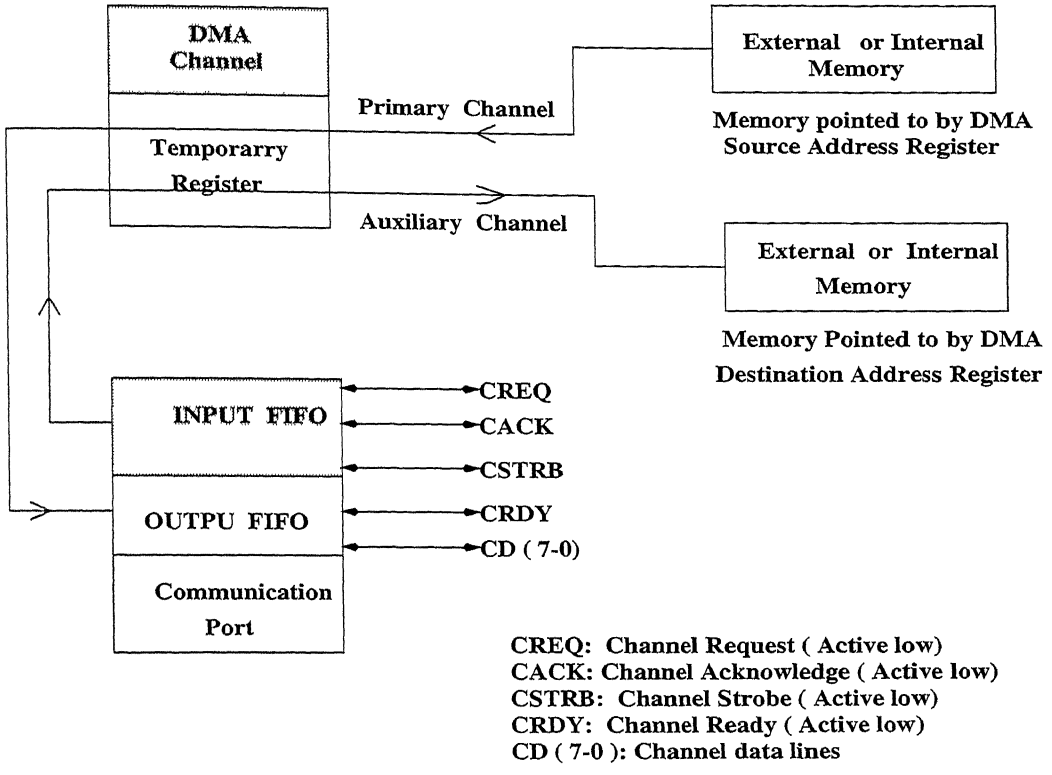
$S^d_{2^{j+1}} f = S^d_{2^j} f * G$

j=j+1

end of while

Appendix B.1

DMA channel arbitration in split mode is described in figure shown below.



As shown in figure there is only one temporary register in each channel. Therefore, a primary channel operation must complete before start an auxiliary channel operation can begin, and vice versa

Appendix C.1

For implementation of wavelet transform on three processors with 256 bin histogram, the histogram is at first divided into three blocks and the number of coefficients in each block is given by,

86 + 85 + 85 = 256 (total)

Now the first block (86 coefficients) is with master processor (TMS320C40 of grabber card) and next two blocks (85 coefficients each) are sent to two slave processors (TMS320C40 of DSP board). Since we used Haar wavelet the number of Haar coefficients is M=2 Wavelet transform implementation technique is explained below:

First level of decomposition:

For first block (Master Processor, TMS320C40 of Grabber Card) :

Number of coefficients after the high pass and low pass decomposition filter (here filter is implemented as convolution) = 86 +2-1= 87.

Number of coefficients after dyadic decimation (even) = 44, all the even coefficients are taken.

For second block (TMS320C40-1 of PPC):

Number of coefficients after high pass and low pass decomposition filtering (filter implemented as convolution)= 85+2-1=86.

Number of coefficients after dyadic decimation (even) = 43, all even coefficients are taken.

For third block (TMS320C40-2 of PPC):

Number of coefficients after high pass and low pass decomposition filtering (filter implemented as convolution)= 85+2-1=86.

Number of coefficients after dyadic decimation (odd) = 43, all odd coefficients are taken.

These intermediate coefficients are stored for further decomposition and sent to *master processor* only after they are decomposed up to final levels.

Second Level of Decomposition:

For first block (Master processor):

Number of coefficients after high pass and low pass decomposition filtering = 44+2-1=45.

Number of coefficients after dyadic decimation (even)= 23, all even coefficients are taken.

For second block (TMS320C40-1).

Number of coefficients after high pass and low pass decomposition filtering = 43+2-1=44.

Number of coefficients after dyadic decimation (odd) = 22, all odd coefficients are taken.

For third block (TMS320C40-2):

Number of coefficients after high pass and low pass decomposition filtering = 43+2-1=44.

Number of coefficients after dyadic decimation (even)= 22, all even coefficients are taken.

Third Level of Decomposition :

For first block (Master Processor):

Number of coefficients after high pass and low pass decomposition filtering = 23 +2-1=24.

Number of coefficients after dyadic decimation (even)= 12, all even coefficients are taken

For second block (TMS320C40-1):

Number of coefficients after high pass and low pass decomposition filtering = 22 +2-1= 23.

Number of coefficients after dyadic decimation (even)= 12, all even coefficients are taken.

For third block (TMS320C40-2):

Number of coefficients after high pass and low pass decomposition filtering = 22+2-1=23.

Number of coefficients after dyadic decimation (odd)= 11, all odd coeffi-

cients are taken.

After decomposition of each blocks separately by all the three processors they are sent to master processor for final coefficients calculations and feature extractions. The technique employed to get final coefficients at all the three levels are shown below

First level final coefficients (total number of coefficients=129) :

44^{th} coefficients of I^{st} block and 0^{th} of 2^{nd} block is overlapped and added i.e.,

0,1,2,3,. . .,42,43

0,1,2,. . .,40,41,42

0,1,2,. . .,40,41,42

Second level final coefficients (total number of coefficients = 65) :

To get the final second level decomposition coefficients 23^{rd} coefficients of I^{st} block and I^{st} coefficients of 2^{nd} block and 22^{nd} coefficients of 2^{nd} block and I^{st} coefficients of 3^{rd} block is overlapped and added, i.e,

0,1,2,. . .,20,21,22

0,1,2,. . .,20,21

0,1,2,. . .,20,21

Third level final coefficients (total number of coefficients = 33) :

To get the the final third level coefficients, 12^{th} coefficients of I^{st} block and I^{st} coefficient of 2^{nd} block and 12^{th} coefficient of 2^{nd} block and I^{st} coefficient of 3^{rd} block is overlapped and added, i.e,

0,1,2,. . .,10,11,12

0,1,2,. . .,10,11,12

0,1,2,. ,10,11

Appendix C.2

In this section we explain in detail the wavelet transform implementation of 256 bins histogram on two processors. The histogram vector is divided in two blocks of equal size, i.e the number of coefficients in each blocks are 128. Three level of decomposition are performed on each block. The interesting part of this implementation scheme is that we don't calculate the final coefficients at every level of decomposition, but the coefficients of all the levels are sent parallely to the master processor for feature extraction. The implementation scheme is explained below:

First Level of Decomposition :

First block (TMS320C40-1) :

Number of coefficients after high pass and low pass decomposition filtering = 128 +2-1=129.

Number of coefficients after dyadic decimation (even) = 65, all even coefficients are taken.

Second block (TMS320C40-2) :

Number of coefficients after high pass and low pass decomposition filtering = 128+2-1=129.

Number of coefficients after low pass and high pass decomposition filtering = 128+2-1=129.

Number of coefficients after dyadic decimation (even) = 65

Second Level of Decomposition :

For the first block (TMS320C40-1) :

Number of coefficients after high pass and low pass decomposition filtering = 65 +2-1= 66.

Number of coefficients after dyadic decimation (even) = 33.

For the second block (TMS320C40-2) :

Number of coefficients after high pass and low pass decomposition filtering = 65+2-1=66.

Number of coefficients after dyadic decimation (odd) = 33.

Third Level of Decomposition :

First block (TMS320C40-1) :

Number of coefficients after high pass and low pass decomposition filtering = 33+2-1= 34.

Number of coefficients after dyadic decimation (even) = 17

Second block (TMS320C40-2) :

Number of coefficients after high pass and low pass decomposition filtering = 33+2-1=34

Number of coefficients after dyadic decimation (even) = 17.

These coefficients are overlapped and added to get the final coefficients values at every level of decompositions and finally features are extracted from these coefficients. The overlap and add scheme for this implementation technique is shown below: First Level Decomposition Coefficients (129 coefficients):

0,1,2,. .,62,63,64

0,1,2,. . .,62,63,64

Second Level Decomposition Coefficients (65 coefficients) :

0,1,2,. . .,30,31,32

0,1,2, . .,30,31,32

Third Level Decomposition Coefficients (33 coefficients) :

0,1,2,. . .,14,15,16

0,1,2,. . .,14,15,16

Bibliography

[1] Anil K. Jain, *Fundamentals of Digital Image Processing* Prentice Hall, 1989

[2] Jae S. Lim *Two Dimensional Signal and Image Processing* Prentice Hall, Enmglewood Cliffs

[3] Julius T. Tou and R. C Gonzalez, *Pattern Recognition Principles* Addison-Wesley Publishing Comopany, 1974.

[4] Richard P. Kruger and William B. Thompson , *"A Technical and Economic Assesment of Computer Vision for Industrial Inspection and Robotic Assembly ,"* Proc. of the IEEE, Vol. 69, No. 12, Dec 1981.

[5] S. K. Chang, C. W. Yan, Donald C. Dimitroff, and Timothy Arndt, *"An Intelligent Database System System,"* IEEE Trans. on Software Engineering, Vol. 14, No. 15, May 1998.

[6] Hon-Son Don, King-Sun Fu, C R Liu and Wei-Chung Lin, *"Metal Surface Inspection Using Image Processing Techniques,"* IEEE Trans. System, Man and Cybernetics, Vol. SMC-14, No.1, Jan. 1984.

[7] G. N. Saridis and D. M. Brandin, *"An Automatic Surface Inspection System for Flat Rolled Steel,"* Automatica, Vol. 15, pp. 505-520, Sept 1979.

[8] R. T. Chin and C. A. Harlow, *"Automated Visual Inspection: A Survey,"* IEEE Trans. Pattern Anal. Machine Intell., Vol. PAMI -4, pp. 557-573, Nov. 1982.

[9] B. R. Suresh, R A. Fundakowski,T. S Levitt and J. E. Overland, *"A Real Time Automated Visual Inspection System for Hot Steel Slabs,"* IEEE Trans. on Pattern Ana. and Machine Intell. Vol. PAMI-5, No. 6, pp. 563-572, Nov. 1983.

[10] F Idris and S. Panchanathan, *"Review of Image and Video Indexing Techniques"*, Journal of Visual Communication and Image Representation, Vol.8, No.2, pp. 146-166,June 1997.

[11] Asha Vellaika and C. C. Jay Kuo, *"Content-Based Image Retrieval Using Multiresolution Histogram Representation,"* Digital Image Storage Archiving Systems 2602, pp. 312-323, October 1995.

[12] James Hafner, Harpreet S. Sawhney, Will Equitz, Myron Flickner, and Wayne Niblack, *" Efficient Color Histogram Indexing for Quadratic Form Distance Function "* IEEE Trans. Patt. Analysis and Machine Intell., Vol. PAMI-17, No. 7, pp. 729-736, July 1995.

[13] D. Brzakovic and N. Vujovic, *"Designing a Defect Classification System: A Case Study* , Pattern Recognition, Vol. 29, No. 8, pp. 1401-1419, (1996).

[14] Stephane G. Mallat, *"Wavelets for Vision,"* Proc. of the IEEE. Vol. 84, No. 4, pp. 604-614, April 1996. bibitemmal-2 Stephne G. Malat, *"A Theory for Multiresolution Signal Decomposition: The Wavelet Representation* IEEE. Tran on Pattern Ana and Machine Intell. Vol. 11, No. 7, pp. 674-693, July 1989.

[15] *TMS320C4X, User's Guide* Texas Instruments, Digital Signal Processing Products

[16] *The Occulus F/64 Frame Grabber,User's Manual*

[17] *SPIRIT-Image, User's Guide,* Sonitech International Inc

[18] *Occulus F/64 DSP Toolkit,* Coreco Inc.

[19] *Occulus Driver DSP Toolkit,* Coreco Computer Vision Products.

[20] ODX Programmer's Toolkit Addendum, Coreco Computer Vision Products.